Experiments in Very Large–Scale Analog Computation

Thesis by
Douglas A. Kerns

In Partial Fulfillment of the Requirements
for the Degree of
Doctor of Philosophy

California Institute of Technology
Pasadena, California
1993
(Defended 30 September 1992)

Copyright © 1993
Douglas Kerns
All rights reserved

Acknowledgments

Thanks to God, who made the world for us all to play in.

Thanks to my parents, who brought me into that world and taught me how to play.

Thanks to my wife, Beth Kerns, who supported me both financially and emotionally through the research and writing, and stayed with me all the way to the end.

Thanks to my advisor, John Hopfield, who gave me the freedom to explore, and to Carver Mead, who acted as unofficial advisor for many of my wanderings.

Thanks to the various administrative support people involved in my life at Caltech, who helped me take care of business: Debbie Chester, Chris Favata, Helen Derevan, Donna Fox.

Thanks to my fellow students, past and present, for discussion and support: Brooke Anderson, Ron Benson, Carlos Brody, Tobi Delbrück, Steve DeWeerth, Dawei Dong, Bhusan Gupta, John Harris, David Kewley, John Lazzaro, John LeMoncheck, David MacKay, Mary Ann Maher, Misha Mahowald, Marcus Mitchell, Andy Moore, Rahul Sarpeshkar, Mass Sivilotti, Grace Tsang, Lloyd Watts (and his family).

Thanks to the following organizations for support of various kinds: the National Science Foundation, DARPA, and the Office of Naval Research for financial support and funding of chip fabrication, MOSIS for administration of chip fabrication, JPL and Tanner Research for summer jobs and the resulting income and engineering experience. I want to especially acknowledge the help of specific individuals in these organizations: Silvio Eberhardt, Raoul Tawel, and Anil Thakoor of JPL, and John Tanner of Tanner Research. Without these organizations, virtually none of my graduate career could have happened.

Abstract

The easy and inexpensive availability of microelectronic prototype fabrication allows us to perform many kinds of experiments in the construction of electronic computational machinery. There has been a recent resurgence in analog computation in various guises: electronic implementations of neural networks, other kinds of neuromorphic circuits, and electronic simulations of various physical systems.

This text documents a set of experiments in analog computation in silicon, and includes a short discussion of the relative advantages of analog *vs* digital computation. The most generally useful result of the work is the development of a set of techniques that allow analog circuits to automatically trim themselves, turning marginal components into devices of good precision.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This text documents my wanderings in the world of analog computation in silicon. I sought to contribute useful circuits, techniques, and ideas to designers of large–scale analog computing engines. My experiments were confined to $2\,\mu$m silicon CMOS technology because of cost and availability, but many of the circuits and techniques can be used with other circuit fabrication technologies. The limit of application of the ideas depends, as always, on the imagination of the reader.

One will find that it is often worth expending the effort to achieve an improved result by applying more sophisticated circuit engineering to a relatively unsophisticated fabrication process. The prize is ready, inexpensive access to fabrication of the resulting circuits. While many of the circuits and subsystems presented in this writing could probably be greatly improved by a tailor–made fabrication process, it will rarely be worth the relatively great expense and effort of tailoring a fabrication process to a particular project. This reliance on circuit techniques, rather than process techniques, is one of the main driving forces behind the work in my thesis.

1.1 Syllabus

Chapter 1 introduces the utility of analog computational circuitry in various contexts.

Chapter 2 makes some comparisons between analog and digital computations, and sets up some pointers for comparing effectiveness and efficiency of various methods of performing a given computation.

Chapter 3 covers some continuous–time filter and delay circuits. These circuits offer extremely low power consumption in the audio–frequency range, and may prove to be useful in front–end processing of speech or sonar information.

Chapter 4 covers a set of techniques for using UV light to manipulate the charge stored on floating MOS circuit nodes. Silicon MOS technology provides virtually indefinite storage of charge on insulated nodes, so techniques for controlling stored charge allow long–term storage of data and trim parameters.

Chapter 5 is a catch–all bin for useful bits and pieces of circuitry developed by necessity or curiosity. The circuits in this chapter are not outstandingly original or unusual, but are straightforward extensions and adaptations of others' work. They are included as archival reference material.

Chapter 6 speculates on future developments which may stem from this and others' work on integrated analog computation. The first section presents ideas about continuous–valued multiwire signal encoding as a possible means to gain the best parts of both analog and digital circuit techniques. The second section contains some thoughts on using autocorrective circuits in microelectronic fabrication processes.

1.2 Why analog?

With all the great advances in digital signal processing and computing technology, one might ask, why bother at all with analog computing? It turns out that analog computing and signal processing offer some real advantages over digital approaches in certain (fairly common) cases. Even the most staunch supporter of DSP must admit that there is no substitute for continuous–time filters for antialias operations at the boundaries of DSP systems. There are, in fact, many more places for analog computational or signal processing subsystems.

1.2.1 Hybrid ICs

The idea of "analog VLSI" is growing in popularity in various places and for various reasons. In telecommunications and consumer electronics, it is appealing to integrate all of the control and signal processing functions of a product or subsystem on a single (potentially low-cost) piece of silicon. This ideal usually involves mixing analog and digital information on the same chip, hence the combination of "analog" and "VLSI." Typical components of such mixed–mode systems are filters, amplifiers, A/D and D/A converters, and occasionally "smart" sensors. Often the input to a circuit is analog, and the output is digital, or vice–versa, so the circuit must behave as a computationally enhanced A/D or D/A converter, perhaps with integrated signal conditioning or processing of some sort. Common examples of this sort of device are found in telecom subscriber interface circuits [6] and "smart" sensors [7].

One will frequently find that references combining "analog" and "VLSI" in the same title actually mean this sort of mixing of analog and digital subsystems on a single substrate, and rarely do they refer to analog computation on any scale larger than a simple filtering or scaling operation. The problems faced by the designer of mixed–mode circuitry have much in common with the designer of truly large–scale analog computational circuitry, but there tends to be the additional complication of crosstalk from digital switching circuitry near sensitive analog circuitry [24]. Virtually all commercial mixed–mode electronics restrict the analog portions of the information processing to simple tasks such as gain and filtering operations, pushing the complicated tasks into a digital circuit of some sort. However, the idea of using analog computation on a large scale or for more complicated processes has recently been gaining momentum, as evidenced in the publication of Mead's text [1].

1.2.2 Analog computing

Historically, analog computers allowed a rapid search of a large parameter space to find singularities and other critical points [14, 15]. While analog computing techniques have been largely neglected in the past decade, the technology now exists to build faster, more compact, and possibly more accurate analog computers than ever before. The same technology that allows us to build high–performance digital computing machines can also be used to build high–performance analog computing machines. The advent of widely available BiCMOS technology allows us to build good operational amplifiers as well as a host of other analog computational circuits, straightforward interfaces to digital computer systems, and even the possibility for integrated programmability from digital host computers [16].

More recent work in analog VLSI has taken advantage of the relatively cheap and easy avaliablility of custom silicon circuit fabrication to build a variety of special–purpose analog computers. These analog computers are not generally called "analog computers," but, rather, they are named for the system to which they are analogous, such as the silicon retina, the electronic cochlea, and so on. Just as in the early days of analog computing, however, these modern analog computers allow real–time exploration of a large space of parameters in the system being modeled.

In addition, there has been some activity in the area of actually using these special–purpose computing engines as pieces of larger systems. This application trend is especially notable in models of brain sensory systems, where immediate applications may be found for good real–world interfaces for computers and robotics [5, 19].

1.2.3 Brain research tools

Analog techniques may also prove to be useful tools in brain research, allowing experiments in "downhill synthesis" [17] as well as providing special-purpose analog computers to emulate real neurons [13]. Examples of synthetic explorations of some merit are found in the vision work of Mahowald and Mead [4], the auditory work of Lyon and Mead [8], Lazzaro [10], and Watts [11], and the work of Ryckebusch in synthesizing various central pattern generators [12].

Such synthetic exploration allows researchers to probe into the hazy area between hardware and software, exploring how various regions of brains might work. Many parts of the brain and many modes of its functioning are inaccessible to current technology, so the ability to synthesize a facile, responsive, and observable model is a great bonus.

1.2.4 Art and science of efficient engineering

There is a strong artistic appeal to using low-level physics directly for a computational task. This aesthetic appeal is complemented by the hard engineering fact of greater efficiency in energy and area; such designs are called "elegant." To further add to the elegance, one finds that thinking in terms of low-level physics can lead to new solutions to old problems, driving the technology into previously unexplored areas, as pointed out by Harris [18]. Some information processing algorithms map very naturally and gracefully onto analog computing hardware, resulting in area– and power–efficient performance. One of the best examples of this is Mead's resistive mesh, which elegantly computes a smoothing function [1]. Extensions of this example can be found in the work of Harris [18] and others [19].

Energy efficiency is becoming a serious consideration in computing machinery, as advances in fabrication and packaging technology pack more and more machinery into each cubic centimeter. Computation costs energy, and the waste heat must be removed from any computing machine in some way. As packing densities increase, removing this waste heat becomes a more and more serious problem. Any method that increases the amount of computation that can be done for a unit of energy also increases the physical density to which we are allowed to pack the computing machinery before it overheats.

It is interesting to consider a comparison between current commercial computing technology, computing technology that is still in the research labs, such as that described in this text, and the best available examples using any computing technology. Our brains are an existence proof of very powerful and efficient computational algorithms for a variety of tasks. A human brain dissipates about 40 W in a volume of about $1500\,\mathrm{cm}^3$, for a power density of about $2.7 \times 10^{-2}\,\mathrm{W/cm}^3$. A standard microprocessor uses roughly 2 W in a volume of about $10\,\mathrm{cm}^3$, for a power density of $0.2\,\mathrm{W/cm}^3$. One order of magnitude doesn't look too bad, if we can ignore the differences in capabilities. However, if we strip off the volume occupied by the package and the mechanical substrate, then the microprocessor is a chunk of silicon about $1\,\mathrm{cm}^2$, of which perhaps a $30\,\mu\mathrm{m}$ thickness is required for the computing devices, wiring, and so on, giving the same power dissipation in a volume of $3 \times 10^{-3}\,\mathrm{cm}^3$, for a power density of $670\,\mathrm{W/cm}^3$. Clearly, the packaging is necessary in order to reduce the system–level power density to a tolerable level. By contrast, special–purpose analog computers such as a electronic cochlea show a power density of $4.2\,\mathrm{W/cm}^3$ using the same technique.

1.3 VLSA

What can be said in general about very large–scale analog computation? As in any analog circuit design, the designer of VLSA circuitry will face the problem of random component variations. In addition, he/she must synthesize a compatible combination of information representations and electronic circuitry. Robust circuit designs and efficient signal representations are essential to successful VLSA design.

1.3.1 Component variations

Variations in component parameters are a well–known part of all electronic design. Component manufacturers have tried to control variations, and design engineers have tried to design robust circuits. So far, most VLSI manufacturing processes have been aimed at digital circuit fabrication, so fine–scale component variations have gone unnoticed. When one tries to use the same process to fabricate very large–scale analog circuitry (VLSA), one finds the variations, and is faced with the old problem of robust design [21].

VLSA integrated circuitry must either have a low intrinsic sensitivity to component variations, or have the capability to automatically trim out such variations. Truly large–scale circuits would be quite impractical and expensive if the variations had to be manually trimmed, each IC requiring many tens or hundreds of trimming operations. Chapter 4 outlines some techniques for designing automatic trimming into VLSA circuitry, and Chapter 6 speculates on a possible approach for reducing component variations at fabrication time.

1.3.2 Representations

Analog computational circuits can be built to use continuous, discrete, and mixed–mode signal representations in units of voltage, current, or charge. Typical digital circuits use only one unit type (voltage, current, or charge) on a fixed number of discrete levels to represent information. Typical analog circuits use all three unit types, usually in a continuous representation. This flexibility with respect to representation of information can give analog circuits the advantage of greatly simplified interface structures to real–world inputs (sensors of various types, whether integrated or off–chip) and outputs (actuators of various types).

Each type of representation has its advantages and typical uses. One task of the VLSA designer is to become familiar with the different possible representations of a quantity in order to choose the best possible representation for a given design.

voltage Voltage is the classical electrical quantity for representation of information, mostly because chemistry and physics have conspired to make voltage sources easy to build, while current or charge sources are considerably more difficult. A fixed voltage represents a fixed charge–carrier energy, hence the ubiquity of electrochemical voltage sources.

A voltage may be broadcast on a conductor to as many locations as needed in a system. Duplication of a voltage is as simple as connecting another piece of wire in the circuit. In a graph representation of an electrical network, a voltage is considered to be a node property.

current Current is a dual quantity to voltage; a current is a property of the edges of a graph representation of an electrical network. A current is a flux of charge carriers, so a sum of currents may be obtained simply by connecting circuit branches.

charge In some devices, such as charge–coupled devices (CCDs), one finds electrical charge to be the representation of information. A charge is a quantity of charge carriers, and is therefore the integral of a current over time. Electrical charge as a representation mode gives the designer access to both an integration with respect to time and a summation.

component	relative area	comment
wire	1	fundamental component in an IC process
capacitor	1–10	(junction of wires)
transistor	1–10	(a special kind of capacitor)
resistor	10^2–10^3	(very long wires)
inductor	10^4–10^5	(very large wire loops)

Table 1.1: **Components available in silicon CMOS:** *Some of the more common analog circuit components are listed. An integrated circuit process also provides the designer with the opportunity to make use of the details of the semiconductor physics to construct devices that are unrealizable as discrete components.*

1.3.3 VLSA design

A typical analog circuit design uses the broadcast of voltage, summation of current, and integration of charge in many different and interesting ways. In addition, the specific physical properties of electrical devices, such as transistors, allow more complex interactions between the various electrical quantities in ways that can be used to accomplish computational tasks.

This text documents an exploration of some circuits and techniques for performing analog computations using silicon CMOS technology. This exploration has shown that the success of a circuit design for a task often hinges on the proper choice of data/signal representation at various stages of the computation. It is important to ensure a continuous smooth flow of compatibly represented information among different subcircuits, or the total system design becomes dominated by sections whose sole purpose is a trivial change of representation.

In addition, the fabrication technology restricts our choice of components. Table 1.1 lists some of the standard circuit elements available in silicon CMOS. Any effective VLSA design must make wise use of the capabilities of the fabrication technology. For example, a circuit including inductances will not generally be area–efficient, because of the tiny inductances available to the IC designer. Capacitances, on the other hand, are readily available. Resistances are often used in traditional analog design, but a no–frills silicon CMOS process has no provision for layers of high resistivity, so resistors become very large, and should be avoided whenever possible.

Chapter 2

Analog vs Digital

Which is the better mode for computation, analog or digital? This question is almost meaningless when asked without context. The answers one finds on close examination are tightly bound to the particular problem at hand, and, especially, dependent on the required definition of "good."

2.1 Performance

The definition of "good" is critical to evaluating the performance of a computing machine. For one problem, precision may be of ultimate importance, while speed is relatively unimportant and power consumption is irrelevant. For another problem, speed may be paramount, while the requirement for precision is quite small. Each measure of performance has its own relationship to the circuit techniques used in the design of a computing machine.

Table 2.1 lists performance measures which are considered important in various contexts. In most cases, only a few measures are important for any particular application, while the rest are unimportant or irrelevant.

In either design paradigm, analog or digital, these various performance measures trade off against each other, often in complex ways. Terms such as "efficiency" usually

power (usually proportional to speed)
size
speed
dollar cost
design time / effort (often a large fraction of dollar cost)
flexibility or (re)programmability
Shannon information capacity (depends on speed and error rate)
error rate
number of trims
reliability wrt component variations and failures

Table 2.1: **Measures of performance for computing machinery**

refer to ratios of competing measures of goodness, and so are higher–level measures. Some of the measures listed above overlap. For instance, high speed usually implies high Shannon capacity, although the information capacity also depends on the error rate. There is a relatively direct competition between high speed and low power dissipation: each decision made, or signal cycle or bit propagated costs at least some minimum quantum of energy [2].

In addition to the confusion over good and evil, one can choose independently whether to work with discrete or continuous signals, and whether to work in discrete time or continuous time. One can find examples of computing machines in all four sections of Figure 2.1. Traditionally, discretized signal representations are called "digital" regardless of the time representation, while continuous signal representations are generally called "analog."

Power density is becoming a concern in the design of computing machinery. Above a critical temperature, integrated device reliability degrades seriously. A critical temperature corresponds to a critical power density. There is perhaps an order of magnitude of variation in critical power density for a given temperature, depending on the details of packaging, but as integration densities increase and processing rates increase, the power density increases in proportion to each. Mead and Conway asserted that a cru-

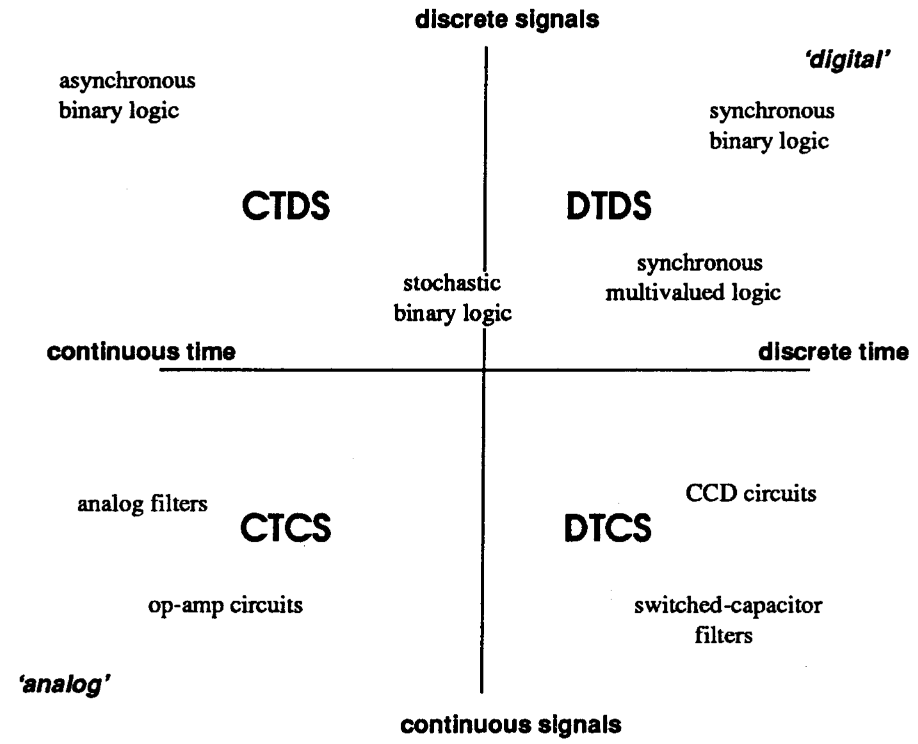


Figure 2.1: **Discretization of quantities:** *The boundary between analog and digital computations is a hazy one, so one needs to be cautious when making comparisons.*

cial measure of computing performance is the unit switching energy [2]. Clearly, as E_{sw} decreases, the total amount of computation that can be performed, in a unit volume and unit time at the critical power density, increases. One might consider that, for any algorithm for a particular task, there is a critical density of computation.

Part of the resurgence in analog computing techniques of various sorts is due to the promise that new computing algorithms may provide more computation for a given energy, hence boosting the critical density of computation. As denser circuit fabrication techniques are developed, the power density limit will become critical in many designs. Examples of circuit technologies that are pushing the power density limit are the chip lamination packaging process developed by Irvine Sensors of Irvine, California, multilayer MOS processes such as that reported by Kioi *et al.* [58], and more conventional multi–chip packaging techniques.

2.2 Scaling

It would be useful, having listed some performance measures, to examine in more detail how analog and digital computations differ along the various axes of performance space. Hopfield showed that it is a straightforward task to discover the scaling laws for various methods of solving a problem [22]. Similarly, Vittoz analyzed analog and digital filters to discover the way in which power dissipation scales with precision [23], and Hosticka presented a simple comparison of CTCS analog, DTCS analog, and DTDS digital circuits based on an information–capacity measure [25]. All three authors discovered very similar results: the algorithmic difference between standard analog and digital techniques for comparable problems give radically different scaling rules for the two modes. Analog circuits tend to be more efficient at low precision, but they scale badly as more and more computational effort is required on a single signal.

Hopfield's analysis centered on a plausible generic electronic technology; the discussion below will use similar methods to examine the real technology of $2\,\mu$m CMOS. A problem that readily admits both analog and digital solutions is that of comparing two quantities and making a decision about which is greater. We can make the comparison between a typical analog implementation and a typical digital implementation for several standard performance measures: area, energy, and time.

Area An analog comparator capable of resolving differences of 30 mV over a range of 4 V can be built in a $2\,\mu$m CMOS technology in an area of about $1500\,\mu\mathrm{m}^2$. The resolution of this comparator is 1 part in 130, or about 7 bits. A 7–bit binary comparator constructed with the same technology will occupy an area of about $10\ 000\,\mu\mathrm{m}^2$. As the precision of comparison scales up, though, the digital circuit begins to win somewhere around 9 or 10 bits, depending on the specific details of the circuits and fabrication process. Trimmable analog circuits can push this tradeoff limit further, to about 12 bits, but the digital circuit still eventually wins, because of its logarithmic scaling. The addition of a single extra bit slice to a digital comparator will double its resolution, while a doubling of resolution of an analog comparator requires approximately a quadrupling of area.

The resolution of the digital comparator is easy to analyze: one simply counts the number of bits in the input word. The resolution R, ratio of the minimum resolvable difference to the maximum representable quantity, is 2^N, where N is the number of bits in each input word to the comparator.

A digital comparator can be built in various ways, but one of the smallest is an iteration of identical bit slices, each of which compares a bit from each input word. The most significant bit passes the result of its comparison to the next comparator slice, and so on. A design of this sort occupies an area that is proportional to N, with a single slice occpying an area of about $1500\,\mu\mathrm{m}^2$ in $2\,\mu\mathrm{m}$ CMOS.

The resolution of an analog comparator is trickier to find. Experimentally, one can find the resolution by looking for the smallest resolvable difference and for the largest representable quantity and taking the ratio. How does the design of the comparator circuit influence these quantities? The largest representable quantity can usually be taken to be the power supply magnitude, or, perhaps somewhat less, in order to account for limitations in the circuit. (A typical differential input stage operates correctly over a limited subrange of the power supply, hence the 4 V range given above, rather than the full 5 V typical power supply). The smallest resolvable difference depends on two properties of the comparator circuit: gain and offset.

The comparator must have sufficient gain to amplify the minimum resolvable difference to an unambiguous decision. We might reasonably state that the gain of the comparator should be sufficient to amplify the minimum resolvable difference to the maximum range, so, then, we require that the comparator gain should be greater than or equal to the resolution R. The gain of a comparator circuit implementation depends on the properties of the transistors, in particular, on the drain conductance. One can find that the gain of an amplifier grows approximately linearly with the length of the transistor channels. In order to maintain a constant bandwidth (or time to complete the computation) the width of the devices also needs to increase, for a net increase in area proportional to the square of the gain increase. In a typical $2\,\mu$m CMOS process, we can expect a gain of about $20\,\mu\text{m}^{-1}$ for a simple output stage, giving an output stage area something like $A_{ao} = (R^2/400)\,\mu\text{m}^2$.

The offset of the comparator limits the smallest resolvable difference in a different way. Even with infinite gain, a comparator with an offset will give an incorrect solution to the comparison problem for some set of inputs. The offset of a comparator circuit depends on matching symmetrical circuit elements. In general, better element matching gives a smaller offset. In a typical comparator circuit, the transistors of the differential input stage are of particular importance to the overall offset. The differential pair will have an offset voltage inversely proportional to the square root of the transistor area [21], so, again, for a given proportion of decrease in offset voltage δV, the circuit area must increase quadratically:

$$\text{Area} = \left(\frac{P_{os}}{\delta V}\right)^2 \quad . \tag{2.1}$$

Based on data in [21], which fits well with informal observations by myself and others here at Caltech, a typical $2\,\mu$m digital CMOS process will give an input offset coefficient P_{os} of about $150\,\text{mV}-\mu\text{m}$. To design a reliable comparator, one should put at least two or three standard deviations into the design. The area of the input stage of the comparator will therefore be $A_{ai} = (R^2/80)\,\mu\text{m}^2$.

The analog comparator will also use a nearly constant area of about $1500\,\mu\text{m}^2$ for wiring, contacts, and minor glue circuitry between the input stage and the output stage. We end up with a total analog comparator area of

$$A_{analog} = 1500 + A_{ai} + A_{ao} = 1500 + \left(\frac{R^2}{80}\right) + \left(\frac{R^2}{400}\right) \quad .$$

We can now directly compare analog and digital comparator areas as functions of the resolution R:

$$A_{analog} = 1500 + \left(\frac{R^2}{80}\right) + \left(\frac{R^2}{400}\right) \qquad A_{digital} = 1500 \log_2 R \quad . \tag{2.2}$$

These area formulas are plotted in Figure 2.2, where the crossover clearly falls near 10 bits.

It is interesting to compare the above derivation of analog comparator area scaling with historical data for the area of DRAM sense amplifiers. Figure 2.3 plots the areas of various DRAM sense amplifiers as a function of the number of bit cells attached to the amplifier. A Dynamic random–access memory depends on capacitive storage of a bit value in a capacitor cell, and on active restoration ("refreshing") of the stored charge with a sense amplifier. The sense amplifier is actually a clocked analog comparator that compares the bitline voltage to some reference voltage to determine whether a "1" or a "0" was stored in the addressed cell. On can roughly measure the resolution of a sense amplifier by determining how many bit cells are attached to a single sense amplifier. The data in the figure are given in square lambda, where lambda is half-linewidth in the technology used. For comparison, lambda is equal to $1\,\mu$m in the $2\,\mu$m technology used for my thesis experiments, so the vertical axis of Figure 2.2 could also read, "square lambda."

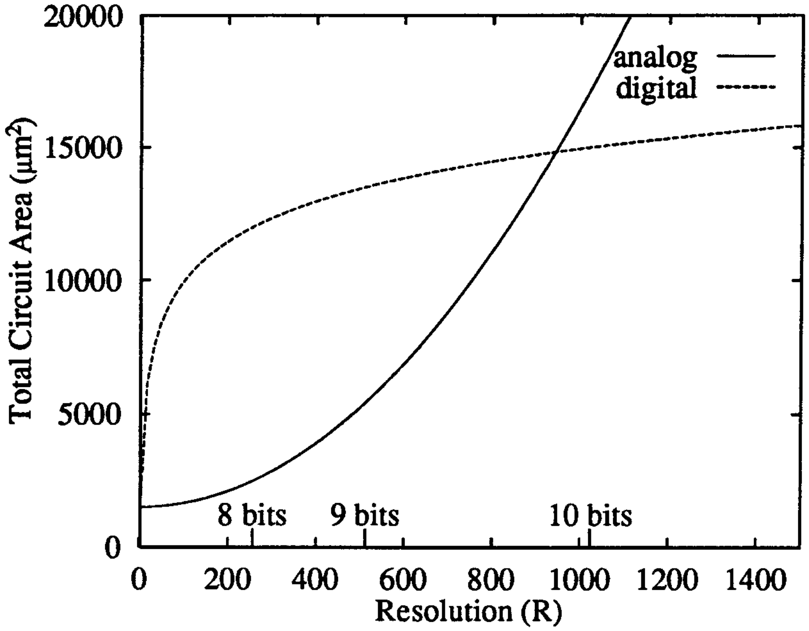


Figure 2.2: **Circuit area** *vs* **precision of comparators:** *Analog comparators begin much smaller than digital comparators, but as the resolution is increased, digital comparators begin to win because of a more efficient representation of the data. Analog comparators scale as a power law, while digital circuits scale logarithmically with resolution.*

The historical data follow a quartic power law, rather than the quadratic power law derived above. The reason for the difference in power law is unclear (keep in mind that the data are from a wide variety of different processes and feature sizes, normalized on the basis of the lithographic process limit, which is assumed to track electrical properties). Nonetheless, it remains that analog comparator areas scale as power law functions of resolution, while digital comparators scale logarithmically.

Because none of the authors directly reported sense amp areas, the data in Figure 2.3 were derived by measurements of the chip photographs presented in the papers. Papers used to obtain the data are reported at the end of the chapter.

Energy The same scaling properties for area carry over to power consumption. For the comparator example, one can analyze the power consumption and discover that the

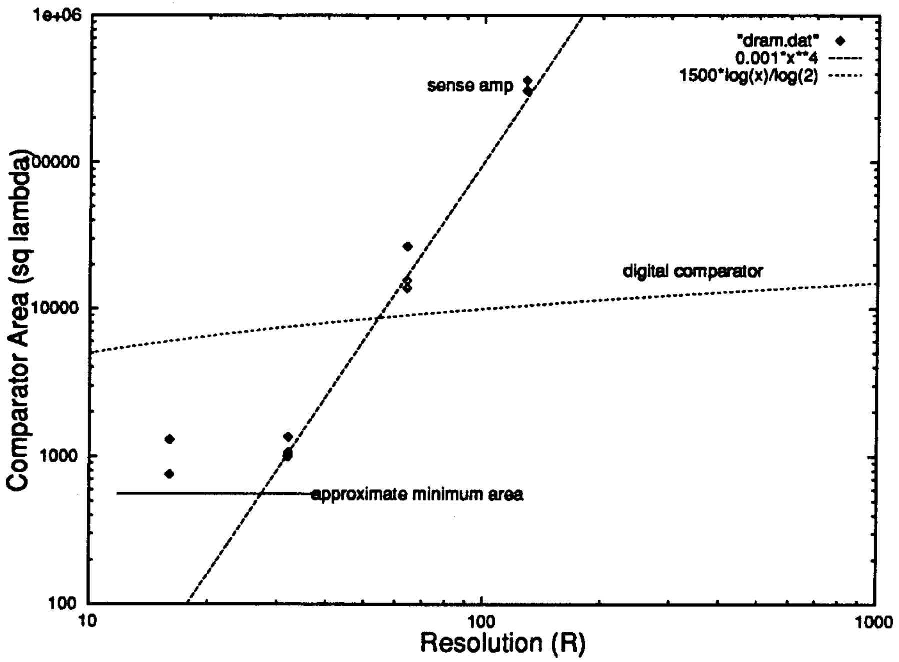


Figure 2.3: **Historical data on DRAM sense amp areas:** *sense amp areas (in square lambdas) scale as the fourth power of the number of bit cells attached to the amplifier. Note the rather sharp lower limit, as circuit area becomes dominated by the constant wiring and contact overhead, rather than transistor properties.*

analog comparator will consume a total energy proportional to square of the precision of the comparison, while the digital comparator will consume a total energy that is logarithmic in the precision. The physics of typical electronic implementations of the comparators dictate that the analog comparator will use less energy than the digital one up to a precision of about six bits.

The digital comparator is composed of N iterated bit slices, each of which must compute two logical functions of its inputs. The inputs to each slice are a pair of bits, one from each of the input words, and two carry signals from the higher–order slices of the comparator. A straightforward implementation of such a slice as a static CMOS logic element uses 22 transistors, 11 of each type. In the worst imaginable case, all of the transistors need to be switched during one comparison, so the energy expended on a comparison operation is about $22NE_0$, where E_0 is the average switching energy of a single device, about 1.5 pJ for 2 μm CMOS if average wiring and contact strays are included. Thus, the total energy cost for a comparison totals up to $E_d = 33N$ pJ.

The analog comparator discussed above consumes a constant bias current I_b during operation. For a typical design, half of the bias current is used to charge and discharge internal nodes of the circuit, and the other half is available for charging and discharging the output load capacitance. The time to complete a comparison can be taken to be the time it takes to charge the output load over half of the power supply range. This time estimate is somewhat pessimistic, as in a typical application, half the power supply range is perhaps an order of magnitude larger than the voltage required to saturate a differential input stage. We can reasonably take the load of the comparator to be the input of another comparator, or a substantially similar circuit.

From the area analysis above, we can find the load capacitance as a function of precision. The input stage requires a total transistor area of about $R^2/400\,\mu\text{m}^2$, of which half will be loading the comparator. The gate capacitance in our example 2 μm CMOS process is about 5×10^{-4} pF per μm^2. These parameters allow us to compute the total energy expenditure for a single comparison operation, with a typical 5 V power supply. The energy expenditure of our analog comparator is

$$E_a = I_b V_{supply} T_{comp} = 2.5 \times 10^{-15} V_{supply}^2 R^2 = 6.3 \times 10^{-14} R^2 \quad . \tag{2.3}$$

In order to make an easy comparison, then, we can compare the analog and digital energy costs as functions of R:

$$E_a = 6.3 \times 10^{-14} R^2, \qquad E_d = 33 \times 10^{-12} \log_2 R \quad . \tag{2.4}$$

These functions are plotted in Figure 2.4. The crossover point occurs a little short of six bits for the designs presented above. More ingenious circuit techniques may push

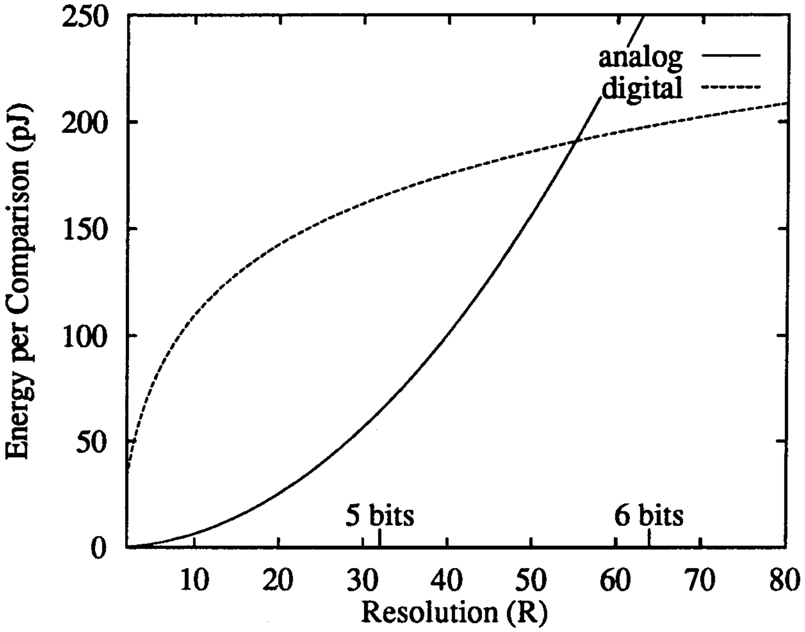


Figure 2.4: **Energy cost** *vs* **precision of comparators:** *Analog comparators are much more efficient at low precision, but as the resolution is increased, digital comparators again win because of a more efficient representation of the data.*

the crossover out as high as ten bits or so with trimming, but the digital comparator will always eventually win as the resolution of the comparison is increased.

In many discrete–time systems, clocked comparators using feedback are used. A classic example is the DRAM sense–amp, which detects a bit value in a dynamic memory cell by making an analog voltage comparison and generating a digital output value. Comparators which use feedback can be shown to be considerably more efficient than the feedforward analog comparator discussed above. The limitation on the use of a fed–back comparator circuit is that it must be reset before each comparison, and so is intrinsically a discrete–time circuit.

As shown in the historical data for DRAM sense amps, analog comparators still suffer from poor scaling rules in comparison with digital comparators.

Time If we try to estimate the time required for the comparison computation, we run into some interesting results. Digital circuits can be pipelined to yield a constant–time operation with a latency which depends on the word length. An analog comparator uses quite a different comparison algorithm, so the analog comparison takes longer and longer to compute as the resolution increases. This increase comes directly from the fact that the analog algorithm for comparison generally comes down to subtracting currents, and, as the difference current shrinks, the time required to charge a node capacitance to make a decision grows proportionally.

The analog/digital comparison in the previous paragraphs hinges on the fact that digital computations are typically performed using a logarithmic code which is distributed over many wires, while analog computations are typically performed with a linear coding scheme on a single wire. There is no apparent fundamental reason not to try to devise nonlinear coding schemes for analog computations; a simple example is the use of log and antilog scaling units to perform analog multiplications [52]. A more sophisticated approach might attempt to beat the SNR limit by distributing a single signal on many wires, as is done in the digital case. See Section 6.1 for more discussion of this idea.

The relative efficiencies of analog and digital circuitry come from the *algorithms* implemented in the hardware, and are not tied directly to the hardware itself. The prefactors between various algorithms are determined by the detailed physics of the particular implementation technology, but the scaling laws are intrinsic to the algorithms. The ingenuity of the circuit engineer allows room for advantage to be taken in both areas: better hardware algorithms will have better scaling laws, and better use of the intrinsic device physics will give better scaling coefficients.

2.3 Example: delays

Vittoz presented a detailed analysis of the theoretical power consumption limits of analog and digital filters in a scalable measure of "power per pole" [23].

Table 2.2 lists several performance parameters of four types of delay lines: continuous–time, continuous–signal (denoted here by the abbreviation CTCS, these circuits are generally considered "analog"); continuous–time, discrete–signal (CTDS); discrete–time, continuous–signal (DTCS); discrete–time, discrete–signal (DTDS, a "digital shift register"). The CTCS line is built of operational transconductance amplifiers and capacitors (a technique often referred to in the literature as OTA–C) in a structure very similar to the silicon cochlea of Lyon and Mead [8]. The CTDS line is a chain of current–limited digital inverters, and is discussed at greater length in Section 3.2.2. The DTCS line is a chain of sample–and–hold sections. The DTDS line is a digital shift register.

Important measures of delay circuit performance are the area occupied, the power dissipated, the total delay from one end to the other, the bandwidth, and the resolution. From these measures, we can compute some standard figures of merit, such as delay–bandwidth product, information capacity, equivalent switching energy, and so on. Because of the different methods of constructing the delay lines, the basic performance measures will have different interdependencies for the different delay lines. Each will be considered in some detail.

CTCS The CTCS delay line is constructed using OTA–C techniques, using Mead's wide–range OTA ([1], Chapter 5). Each stage of the line occupies an area of $(152 \times 62)\,\mu\text{m}^2$, for a total area of $367\,536\,\mu\text{m}^2$ for the 39 stages in the test circuit. The power dissipation of the CTCS delay line varies according to the bias currents of the OTAs. In the subthreshold range, the transconductance of an OTA is directly proportional to the bias current, so there should be a linear relationship between the power dissipation and the bandwidth. Because of the linear filtering operation of each stage, the CTCS delay line should show a constant delay–bandwidth product, giving a linear inverse relationship between power dissipation and total delay. Experimentally, measurements from the CTCS delay line diagrammed in Figure 3.15 show that there is indeed a linear interdependence among power dissipation, bandwidth, and delay time, with coefficients as noted in Table 2.2. In the subthreshold range, the OTAs will operate correctly with signals up to 100 mV peak–to–peak, with a noise floor of about 1 mV.

CTDS The CTDS delay line is constructed of a chain of circuits like that diagrammed in Figure 3.20. These delay cells are much like digital CMOS inverters with the addition of current–limiting devices to introduce a controllable transition delay. The CTDS circuit is discussed in more detail in Section 3.2.2. The CTDS delay circuit propagates binary signals in continuous time. The resolution of a single delay line is trivially $R = 2$ distinct signal values. Each section of the line occupies an area of $(145 \times 148)\,\mu\mathrm{m}^2$, for a total area of $579\ 420\,\mu\mathrm{m}^2$ for the 27 stages of the test circuit. The power dissipation of the CTDS delay circuit diagrammed in Figure 3.20 is determined partly by the data flowing through it, and partly by the bias currents in the devices. A signal transition on the input costs a unit switching energy due to the capacitive charging of the transistor gates, regardless of the time required to effect the transition. The bias circuit wastes a steady–state current proportional to the current limit of the delay device. The gain stages cost more switching energy units, as well as wasting some current during switching, proportional to the switching transition period. The total power dissipation follows the relation

$$P_{total} = V_{dd} I_{ss} = V_{dd} \left[I_b + \frac{A f_d}{I_b} \right] \quad . \tag{2.5}$$

where $A = 0.06\,\mu\mathrm{A}^2/\mathrm{Hz}$, and f_d is the instantaneous frequency of the incoming data stream (half the number of edges per second).

Again, we would expect a linear dependence of the maximum transition rate on the bias current, and a constant delay–bandwidth product. The bandwidth can be taken as half the maximum edge rate, by Nyquist's theorem. Experiments bear out this expectation, with the bandwidth varying as $B \approx 4 \times 10^8 I_b$

DTCS The DTCS delay line is a chain of sample–and–hold sections. Each section occupies an area of $(200 \times 113)\,\mu\mathrm{m}^2$, for a total area of $632\,800\,\mu\mathrm{m}^2$ for 28 stages. The power dissipation of the delay line has two sources: the clock generation circuitry and the buffer amplifiers. The clock generation circuitry is digital switching elements, with a power dissipation proportional to the sampling rate, while the buffer amplifiers are biased with constant current sources. The clocked portions of the circuit consume a current of $I_{clock} = 1.2 f_s\,\mathrm{nA}$ at $V_{dd} = 5\,\mathrm{V}$. The bandwidth of the DTCS delay line is determined by the sample rate at low sample rates, and by the transconductance of the buffer amplifiers at high sample rates. For any given bias current, the buffer amplifiers have a particular cutoff frequency. Measurements show that the buffer amplifiers must consume a supply current proportional to the cutoff frequency f_c: $I_b = 0.8 f_c\,\mathrm{nA}$. The total current consumption is thus

$$I_{ss} = I_{clock} + I_b = 1.2 \times 10^{-9} f_s + 0.8 \times 10^{-9} f_c \text{ amperes} \quad . \tag{2.6}$$

The total delay is dependent on the sampling frequency, provided that the buffer amplifier bandwidth is not exceeded. For the 28 stages of sample–and–hold, the total delay is therefore $28/f_s$. The buffer amplifiers operate correctly over a range of about 4 V. Switching noise dominates the noise floor, and depends partly on the sampling frequency. Experiments show a switching noise component of $V_n = 12.7 + 7.2 \times 10^{-5} f_s\,\mathrm{mV}$

rms, so the resolution is $R = 4000/(12.7 + 7.2 \times 10^{-5} f_s)$.

DTDS The DTDS delay line is a digital shift register. The register propagates binary dignals in discrete time, so its resolution is trivially $R = 2$. Each unit of the register occupies an area of $(50 \times 89)\,\mu\text{m}^2$, for a total area of $178\ 000\,\mu\text{m}^2$ for 40 stages. The power dissipation of the shift register is strongly dependent on the data flowing through the register, as each bit transition costs a unit switching energy. In the simple case of a single bit propagating through the register, the switching energy was measured at 48 pJ for $V_{dd} = 5\,\text{V}$. This measurement can be extended to the typical case of half of the register elements seeing transitions, for an estimated typical power dissipation of $4.8 f_s\,\text{nW}$. The total delay through the register is dependent on the sample clock rate. Each bit moves one stage per clock cycle, so the total delay is $40/f_s$. The shift register propagates bits at a rate of f_s, which can reasonably be considered to be half the bandwidth of the register, by the Nyquist sampling theorem.

CCD Another type of DTCS delay line exists, in essentially the same technology. Transistor channels can be arranged to form a chain, generally known as a charge–coupled device (CCD). When the CCD gates are clocked with the proper waveform sequence, channel charge is transferred from one gate to the next along the chain. For a generic CMOS process, the charge transfer efficiency of a surface–channel CCD is fairly low, between 0.9990 and 0.9999, giving a total charge loss of a few percent along a chain of a length of 50. This means that the CCD delay line is good for about 7 bits. Except for input and output interface devices, a CCD delay line is a passive device. Some of the signal energy is dissipated in the channel resistance (as charge loss), and all other energy expenditure is in the clock circuitry. Therefore, we could possibly make the statement that a CCD delay line dissipates no energy, but, realistically, we must include the clock dissipation in the calculations. For a unit gate area of $400\,\mu\mathrm{m}^2$ in our $2\,\mu$m CMOS technology, our hypothetical CCD delay line will consume an average supply current of about $10^{-11}\ f_s$. The numbers quoted in Table 2.2 for a CCD delay line are estimates, rather than experimental measurements.

A careful examination of the various delay lines reveals that they each have certain advantages. The continuous–signal lines manage to pack much more resolution into a single stage than do the discrete–signal lines. The continuous–time lines tend to have rather low total information capacity as compared to their discrete–time versions. The power dissipation of the CTCS line is virtually constant, independent of the data, while, at the other extreme, the power dissipation of the DTDS line is almost entirely data dependent. Clearly, the advantage does not belong squarely in any one of the four bins. The VLSA designer must carefully weigh the requirements of the task and choose the most suitable circuit technique.

Moreover, one can see that there are several examples of very different ways of using the same technology to accomplish essentially the same task. This variety of techniques was generated by the imaginations of various designers, and there may well be other possible methods. Creativity and imagination are important engineering tools.

CTCS	
unit area:	$9\ 424\,\mu\mathrm{m}^2$
unit delay:	$(2.4 \times 10^{-10}/I_b)$ seconds
bandwidth:	$(1.6 \times 10^{10} I_b)$ Hz
resolution (R):	100
supply current:	I_b
CTDS	
unit area:	$21\ 460\,\mu\mathrm{m}^2$
unit delay:	$1.1 \times 10^{-8}/I_b$ seconds
bandwidth:	$3.73 \times 10^{8} I_b$ Hz
resolution:	2
supply current:	$I_b + (6 \times 10^{-14} f_d/I_b)$ amperes
DTCS:	
unit area:	$22\ 600\,\mu\mathrm{m}^2$
unit delay:	$1/f_s$
bandwidth:	$\min\{f_s/2\ ,\ 1.3 \times 10^{9} I_b\}$ Hz
resolution:	$4\ 000/(13 + 7.2 \times 10^{-5} f_s) \leq 308$
supply current:	$I_b + 1.2 \times 10^{-9} f_s$ amperes
DTDS	
unit area:	$4\ 450\,\mu\mathrm{m}^2$
unit delay:	$1/f_s$
bandwidth:	$f_s/2$
resolution:	2
supply current:	$9.6 \times 10^{-10} f_s$ amperes (average)
CCD	
unit area:	$400\,\mu\mathrm{m}^2$
unit delay:	$1/f_s$
bandwidth:	$f_s/2$
resolution:	100
supply current:	$10^{-11} f_s$ amperes (average)

Table 2.2: **Delay line comparison:** *four types of delay lines, corresponding to the four regions of Figure 2.1, are compared on the basis of several easily accessible performance measures: area, delay, bandwidth, resolution, and power supply current at 5 V. Standard figures of merit can be easily computed from these measures. Estimates of CCD delay line performance are also included.*

Papers used for DRAM sense amp data

W. M. Regitz, J. A. Karp
Three-Transistor–Cell 1024–Bit 500 ns MOS RAM
IEEE Journal of Solid-State Circuits, vol.SC-5, no.5, pp.181–186 October 1970.

R. A. Abbott, W. M. Regitz, J. A. Karp
A 4K MOS Dynamic Random–Access Memory
IEEE Journal of Solid-State Circuits, vol.SC-8, no.5, pp.292–298 October 1973

H. J. Boll, W. T. Lynch
Design of a High–Performance 1024–b Switched Capacitor p–Channel IGFET Memory Chip
IEEE Journal of Solid-State Circuits, vol.SC-8, no.5, pp.310–318 October 1973

C. N. Ahlquist, J. R. Breivogel, J. T. Koo, J. L. McCollum, W. G. Oldham, A. L. Renninger
A 16 384–Bit Dynamic RAM
IEEE Journal of Solid-State Circuits, vol.SC-11, no.5, pp.570–574 October 1976

K. Itoh, K. Shimohigashi, K. Chiba, K. Taniguchi, H. Kawamoto
A High–Speed 16–kbit n-MOS Random–Access Memory
IEEE Journal of Solid-State Circuits, vol.SC-11, no.5, pp.585–590 October 1976

E. Arai, N. Ieda
A 64–kbit Dynamoc MOS RAM
IEEE Journal of Solid-State Circuits, vol.SC-13, no.3, pp.333–338 June 1978

T. Wada, O. Kudoh, M. Sakamoto, H. Yamanaka, K. Nakamura, M. Kamoshida
A 64K x 1 Bit Dynamic ED–MOS RAM
IEEE Journal of Solid-State Circuits, vol.SC-13, no.5, pp.600–606 October 1978

T. Wada, M. Takada, S. Matsue, M. Kamoshida, S. Suzuki
A 150 ns, 150 mW 64K Dynamic MOS RAM
IEEE Journal of Solid-State Circuits, vol.SC-13, no.5, pp.607–611 October 1978

M. Kondo, T. Mano, F. Yanagawa, H. Kikuchi, T. Amazawa, K. Kiuchi, N. Ieda, H. Yoshimura
A High Speed Molybdenum Gate MOS RAM
IEEE Journal of Solid-State Circuits, vol.SC-13, no.5, pp.611–616 October 1978

K. Natori, M. Ogura, H. Iwai, K. Maeguchi, S. Taguchi
A 64 kbit MOS Dynamic Random Access Memory

IEEE Journal of Solid-State Circuits, vol.SC-14, no.2, pp.482–485 April 1979

J. Y. Chan, J. J. Barnes, C. Y. Wang, J. M. deBlasi, M. R. Guidry
A 100ns 5V Only 64K × 1 MOS Dynamic RAM
IEEE Journal of Solid-State Circuits, vol.SC-15, no.5, pp.839–846 October 1980

M. Taniguchi, T. Yoshihara, M. Yamada, K. Shimotori, T. Nakano, Y. Gamou
Fully Boosted 64K Dynamic RAM with Automatic and Self-Refresh
IEEE Journal of Solid-State Circuits, vol.SC-16, no.5, pp.492–498 October 1981

C. A. Benevit, J. M. Cassard, K. J. Dimmler, A. C. Dumbri, M. G. Mound, F. J. Procyk, W. Rosenzweig, A. W. Yanof
A 256K Dynamic Random Access Memory
IEEE Journal of Solid-State Circuits, vol.SC-17, no.5, pp.857–862 October 1982

Chapter 3

Filters and Delays

Time–domain information processing is crucial to almost all engineering and control tasks. Filter theory has been developed in order to address this issue in its most common guise: separating a time–varying "signal" of interest from obscuring "noise" or "interference." The difference between a "filter" and some sort of "analog computer" is often one of intent and interpretation, rather than functionality. In this section, I will discuss two particularly useful analog building blocks: (1) continuous–time analog filters, and (2) continuous–time delay elements for both analog and digital signals.

3.1 CT filters

Continuous–time filters are indispensable blocks, even in a world dominated by digital signal processing. Most real–world signals originate as electrical analogs of continuously varying signals such as air or water pressure, flow rate, temperature, etc., and the end output quantities are also often continuous. Continuous–time filters are required at the interface between continuous–time and discrete–time signals, in order to prevent frequency aliasing. Continuous–time filters can also yield a substantial savings in circuit area and power cost, as discussed in section 2.3.

As a tool for understanding more general integrated filter circuits, I will analyze a

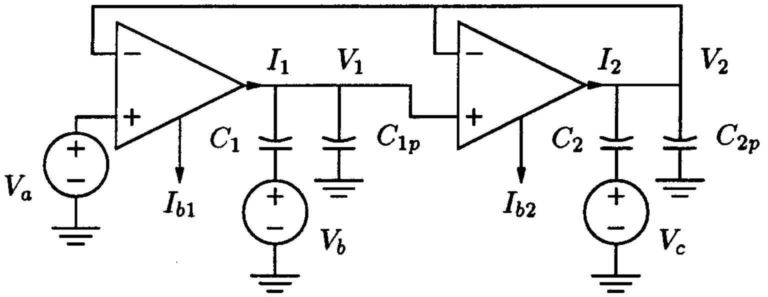


Figure 3.1: **DIFF2 filter circuit:** *explicit capacitance C_1 and parasitics C_{1p} and C_{2p} are included*

particular filter in great detail. Parts of this section were also published in Caltech CNS Memo 12 (1991). The basic circuit design is from Mead and his research group [1], and my primary contribution in this section is the analysis and use of the circuit as a filter, and the extension of the filter analysis to a related four–transistor circuit.

3.1.1 The DIFF2 filter

Mead's DIFF2 circuit ([1], pp.169–173) shows resonant behavior due to capacitance on the "output" node. This resonant behavior can be used for second–order filtering operations if the capacitance is explicitly designed into the circuit, as shown in Figure 3.1. This circuit contains the minimum number of components necessary to realize a second–order filter, although an additional active element may be added in order to de–stabilize the circuit [32].

The DIFF2 filter is a member of the general family of OTA–C filters (constructed of Operational Transconductance Amplifiers and Capacitors), which are particularly well–suited to integrated circuit realization of continuous–time filters, as they can be constructed solely of transistors and capacitors.

Simple linear analysis

A typical OTA has a transfer function somewhat like,

$$I_{out} = I_0 \tanh\left(\frac{\kappa\Lambda_T}{2}(V_+ - V_-)\right) \quad . \tag{3.1}$$

For sufficiently small differential input voltages, we can approximate the amplifier's transfer function with a linear function:

$$I_{out} = g_0 (V_+ - V_-) \quad , \tag{3.2}$$

where $g_0 = \frac{\kappa\Lambda_T}{2} I_0$.

Such an approximation allows us to do a very simple linear analysis of the DIFF2 filter circuit of Figure 3.1. The basic circuit equations are:

$$I_1 = g_1(V_a - V_2) = C_1(\dot{V}_1 - \dot{V}_b) + C_{1p}\dot{V}_1 \tag{3.3}$$

$$I_2 = g_2(V_1 - V_2) = C_2(\dot{V}_2 - \dot{V}_c) + C_{2p}\dot{V}_2 \tag{3.4}$$

or, using Heaviside operational calculus,

$$I_1 = g_1(V_a - V_2) = sC_1(V_1 - V_b) + sC_{1p}V_1 \tag{3.5}$$

$$I_2 = g_2(V_1 - V_2) = sC_2(V_2 - V_c) + sC_{2p}V_2 \quad . \tag{3.6}$$

Notice that I have included the parasitic capacitors C_{1p} and C_{2p} in the analysis. There are two reasons for this: first, these parasitics are unavoidably present in any real circuit; second, there are some interesting and potentially useful side effects due to their presence.

A straightforward derivation yields the following result:

$$V_1(s^2 + \tilde{\omega}_2 s + \tilde{\omega}_1\tilde{\omega}_2) = \frac{C_1}{\tilde{C}_1} sV_b(s + \tilde{\omega}_2) + \tilde{\omega}_1 V_a(s + \tilde{\omega}_2) - \tilde{\omega}_1\left(\frac{C_2}{\tilde{C}_2}\right) sV_c \tag{3.7}$$

$$V_2(s^2 + \tilde{\omega}_2 s + \tilde{\omega}_1\tilde{\omega}_2) = \left(\frac{C_2}{\tilde{C}_2}\right) s^2 V_c + \tilde{\omega}_2 \left(\frac{C_1}{\tilde{C}_1}\right) s V_b + \tilde{\omega}_1\tilde{\omega}_2 V_a \quad , \tag{3.8}$$

where the following symbols have been defined for brevity:

$$\tilde{C}_1 = C_1 + C_{1p} \quad \tilde{C}_2 = C_2 + C_{2p} \quad \omega_1 = \tfrac{g_1}{C_1} \quad \tilde{\omega}_1 = \tfrac{g_1}{\tilde{C}_1} \quad \omega_2 = \tfrac{g_2}{C_2} \quad \tilde{\omega}_2 = \tfrac{g_2}{\tilde{C}_2} \quad .$$

Notice that the forms and coefficients of the left-hand sides of both equations are identical. This is not surprising, since the circuit is the same in both cases, and it has a single resonant frequency and damping coefficient. These can be found by substitution into the canonical second-order ODE form:

$$V(s^2 + 2\xi\omega_0 s + \omega_0^2) = F(s) \tag{3.9}$$

and so we find

$$\omega_0 = \sqrt{\tilde{\omega}_1\tilde{\omega}_2} \qquad \xi = \sqrt{\frac{\tilde{\omega}_2}{\tilde{\omega}_1}} \qquad \text{or,} \qquad Q = \frac{1}{2\xi} = \sqrt{\frac{\tilde{\omega}_1}{4\tilde{\omega}_2}} \quad . \tag{3.10}$$

An important feature to notice is that the damping coefficient ξ is positive for all achievable values of the circuit parameters. This means that this circuit should be unconditionally stable; we pursue a more detailed discussion of this issue in the detailed linear analysis section.

Various choices of the input nodes and a choice of either node 1 or node 2 as the output change the positions of the filter zeros and hence the overall behavior of the filter. It is possible to realize lowpass, bandpass, highpass, and bandstop characteristics with this circuit, merely by choosing the connections appropriately, so the DIFF2 circuit can be considered a general–purpose filter module.

Table 3.1 lists the possible voltage-in, voltage-out transfer function numerators for various configurations of the DIFF2 filter. The voltage-in, voltage-out transfer function for the filter in any configuration is of the form

$$H(s) = \frac{N(s)}{D(s)} \qquad \text{where} \qquad D(s) = (s^2 + \tilde{\omega}_2 s + \tilde{\omega}_1\tilde{\omega}_2)$$

input node(s)	output	numerator $N(s)$	*comments*
V_a	V_2	$\tilde{\omega}_1\tilde{\omega}_2$	lowpass
V_b		$\tilde{\omega}_2\left(\frac{C_1}{\tilde{C}_1}\right)s$	bandpass
V_c		$\left(\frac{C_2}{\tilde{C}_2}\right)s^2$	highpass
V_a, V_b		$\tilde{\omega}_2\left(\frac{C_1}{\tilde{C}_1}\right)(s+\omega_1)$	
V_a, V_c		$\left(\frac{C_2}{\tilde{C}_2}\right)\left(s^2+\tilde{\omega}_1\omega_2\right)$	bandstop
V_b, V_c		$\tilde{\omega}_2\left(\frac{C_1}{\tilde{C}_1}\right)(s+\omega_1)$	
V_a, V_b, V_c		$\left(\frac{C_2}{\tilde{C}_2}\right)\left(s^2+\left(\frac{C_1}{\tilde{C}_1}\right)\omega_2+\tilde{\omega}_1\omega_2\right)$	
V_a	V_1	$\tilde{\omega}_1\left(s+\tilde{\omega}_2\right)$	DIFF2
V_b		$\left(\frac{C_1}{\tilde{C}_1}\right)\left(s^2+\tilde{\omega}_2 s\right)$	
V_c		$-\tilde{\omega}_1\left(\frac{C_2}{\tilde{C}_2}\right)s$	bandpass
V_a, V_b		$\left(\frac{C_1}{\tilde{C}_1}\right)\left(s^2+\left(\tilde{\omega}_1+\tilde{\omega}_2\right)s+\omega_1\tilde{\omega}_2\right)$	
V_a, V_c		$\tilde{\omega}_1\left(1+\left(\frac{C_2}{\tilde{C}_2}\right)\right)\left(s+\tilde{\omega}_2\left(\frac{\tilde{C}_2}{C_{2p}}\right)\right)$	
V_b, V_c		$\left(\frac{C_1}{\tilde{C}_1}\right)\left(s^2+\left(\tilde{\omega}_2-\omega_1\left(\frac{C_2}{\tilde{C}_2}\right)\right)s\right)$	
V_a, V_b, V_c		$\left(\frac{C_1}{\tilde{C}_1}\right)\left(s^2+\left(\tilde{\omega}_2+\omega_1\left(1-\left(\frac{C_2}{\tilde{C}_2}\right)\right)\right)s+\omega_1\tilde{\omega}_2\right)$	

Table 3.1: **Table of Available Numerators:** *Numerators of transfer functions for different voltage–in, voltage–out configurations of the DIFF2 filter. Unspecified input nodes should be tied to appropriate constant voltages.*

and $N(s)$ is a polynomial from table 3.1. The DIFF2 filter is not quite a biquad filter, as zeros cannot be placed arbitrarily, but it can implement many useful filter functions.

There are, of course, more possibilities for using these circuits than just as *voltage–* mode devices, as currents can also be used as the signals.

Figures 3.2 through 3.5 show data taken in the small-signal linear range of operation. In order to clearly show the various filter functions at this low amplitude, the noise floor of the data was reduced about 20 dB by averaging.

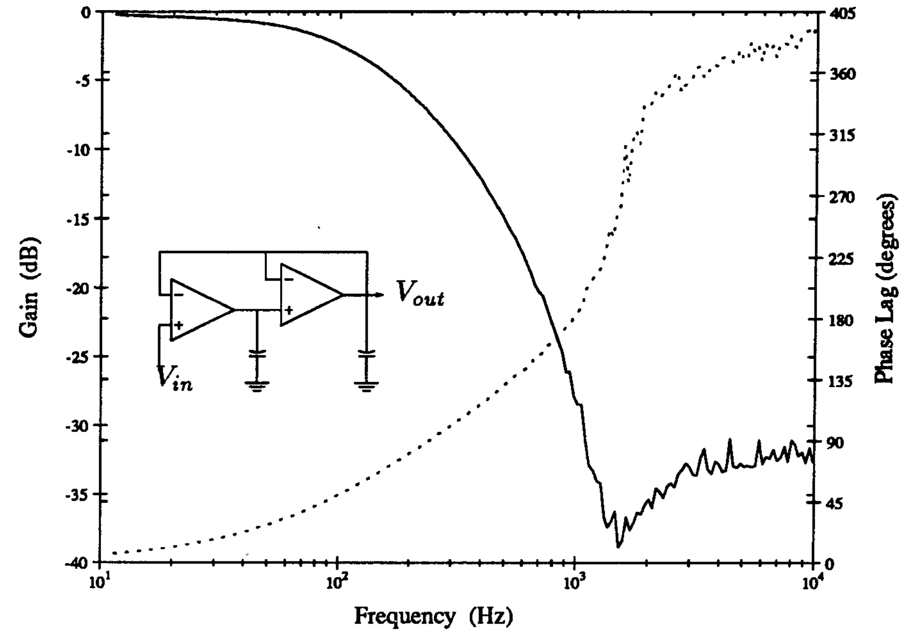


Figure 3.2: **DIFF2 circuit in lowpass configuration**

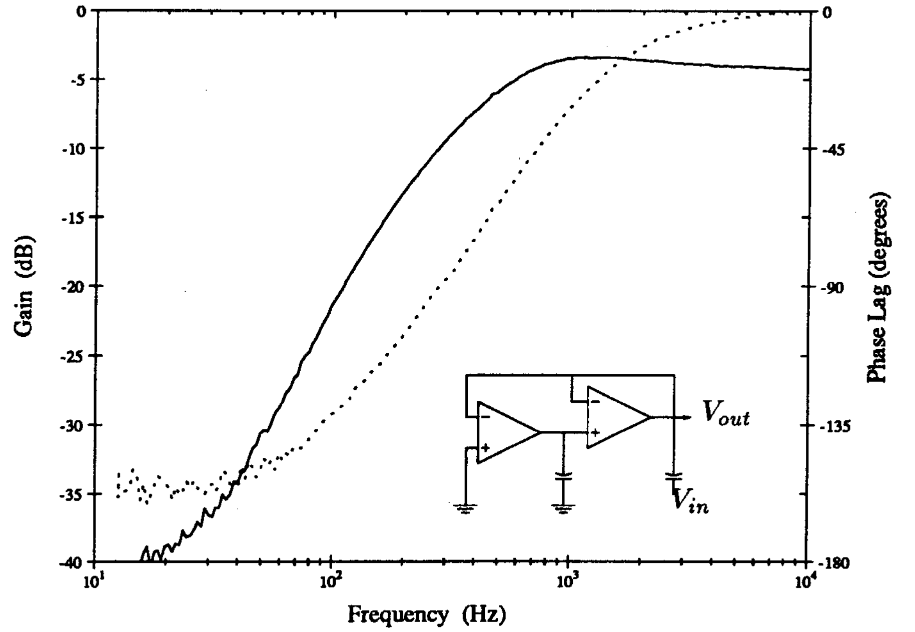


Figure 3.3: **DIFF2 circuit in highpass configuration**

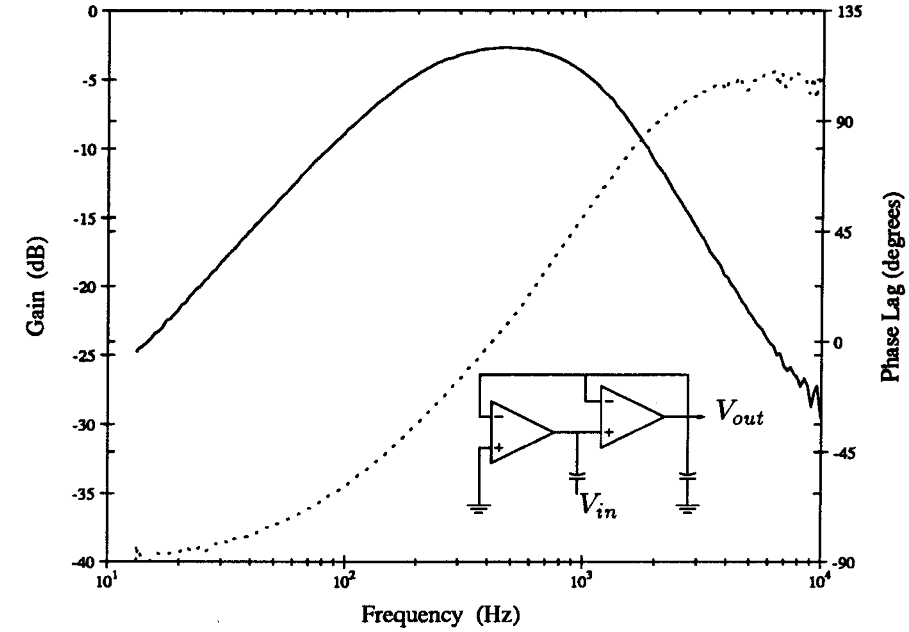


Figure 3.4: **DIFF2 circuit in bandpass configuration**

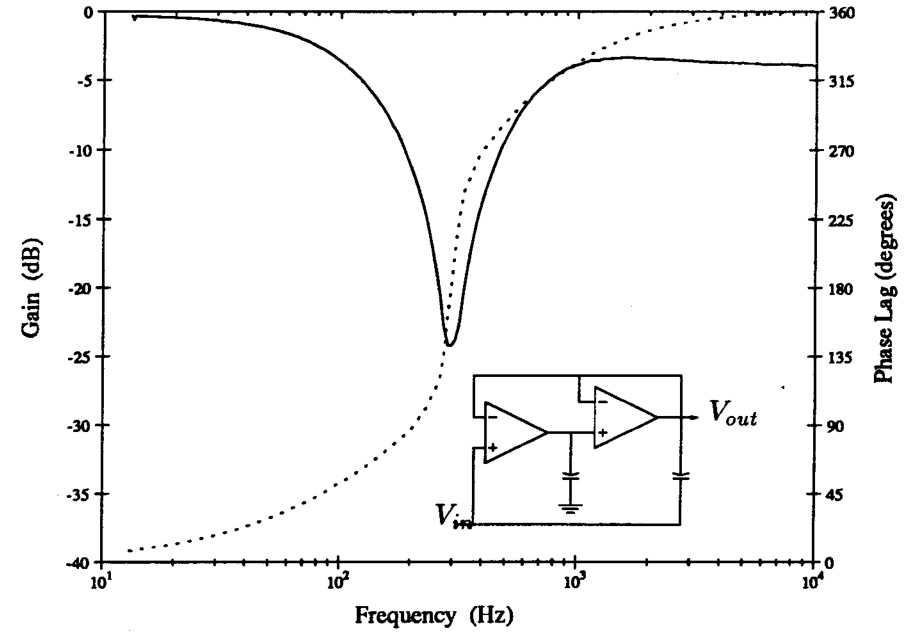


Figure 3.5: **DIFF2 circuit in bandstop configuration**

All of the data curves show a slight distortion at the high-frequency end which can be attributed to the on-chip buffer amplifier's limited bandwidth. The low-pass filter data shows a rebound above 1 KHz; this may be due to stray feedthrough capacitance in various parts of the instrumentation circuitry. The significant difference in amplitude between the low- and high-frequency ends of the bandstop gain curve is due to the presence of the C_{2p} parasitic capacitance.

Detailed linear analysis

Typical transconductance amplifiers are moderately complicated circuits with several internal nodes, each of which has a small but finite parasitic capacitance to ground and to the other circuit nodes. These parasitics give rise to higher-order behavior in the filter circuit which can lead to instability under certain bias conditions. Standard texts on analog design, such as Gray and Meyer [28], have elaborate transistor models based on the standard small-signal linearization technique, taking account of many details in the behavior of a transistor. Such detail is necessary for designs that push the limits of the technology, but a simple zero-order model of a MOS transistor as a voltage-controlled current source will suffice to illustrate the problems one might encounter with internal amplifier nodes.

For simplicity, let's begin by using the simple OTA diagrammed in Figure 3.6(a); the DIFF2 is thus a fourth-order circuit, as diagrammed in Figure 3.6(b).

Substitution for the circuit parameters gives, after some work,

$$\begin{aligned} D(s) \;=\; & s^4 + s^3\left[\alpha\tilde{\omega}_1 + (\alpha + \frac{1}{2})\tilde{\omega}_2\right] + s^2\tilde{\omega}_2\left[(\alpha^2 + \frac{\alpha}{2} + \frac{1}{4})\tilde{\omega}_1 + \alpha\tilde{\omega}_2\right] \\ & + s\tilde{\omega}_1\tilde{\omega}_2\left[\alpha(\alpha + \frac{1}{2})\tilde{\omega}_2 + \frac{\alpha}{2}\tilde{\omega}_1\right] + \alpha^2\tilde{\omega}_1^2\tilde{\omega}_2^2 \quad , \end{aligned} \tag{3.11}$$

where α is the ratio of the intended capacitance to the parasitic capacitance.

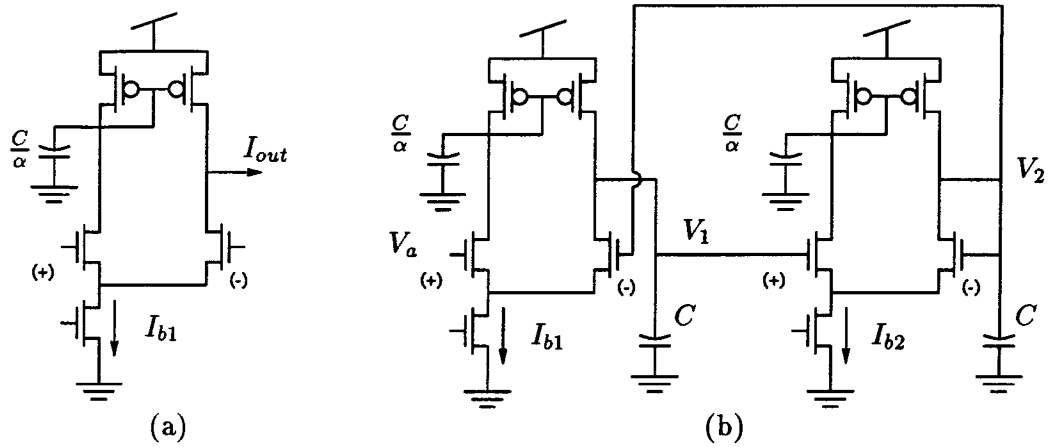


Figure 3.6: **A parasitic capacitance in the DIFF2 circuit:** *(a) a simple OTA circuit (after Mead[1], chapter 5), and (b) a DIFF2 using this circuit*

The key question for stability is whether the zeros of D have negative real parts. To address this, a possible approach is to use the Routh–Hurwitz method (see any control theory text, for example, [27] for more details on this method). The Routh–Hurwitz method gives the following criterion for stability:

$$c_1(g_1, \alpha) = \frac{a_1 a_2 a_3 - a_3^2 - a_1^2 a_4}{a_1 a_2 - a_3} > 0 \quad , \tag{3.12}$$

where the a_i are the coefficients of D:

$$D(s) = s^4 + a_1 s^3 + a_2 s^2 + a_3 s + a_4 \quad . \tag{3.13}$$

If we choose $\tilde{\omega}_2 = 1$ for scale, then Equation (3.12) gives a function of α and g_1 that lets us know the regions in which the circuit is stable.

We have assumed two identical OTA-C sections (hence a single parameter α), and have made use of the fact that all the time constants associated with a single amplifier are related by the capacitor sizes, because our simple OTA has a single bias current.

It is clear that symbol-manipulation tools such as Mathematica can be very valuable in detailed analysis of even relatively small and simple circuits. Many commercial circuit simulators also perform numerical small-signal analyses, but symbolic analysis often benefits the designer by pointing out critical boundaries and sensitivities.

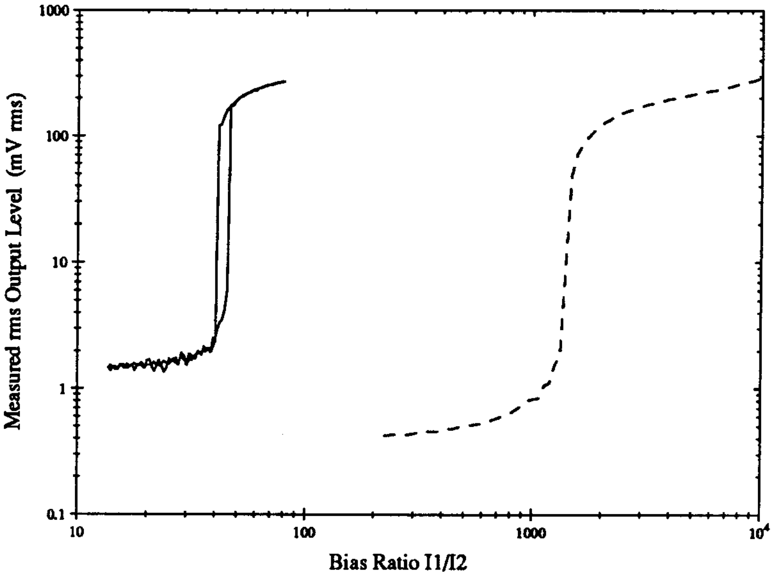


Figure 3.7: **DIFF2 circuit instability due to parasitics:** *a plot of measured rms voltage on node V_1* vs *the amplifier transconductance ratio $\frac{g_1}{g_2}$ for DC inputs (no signal in). The solid curves are the stability limits for a filter with relatively large parasitics due to sloppy layout, and the dashed curve shows the limit for the same circuit with smaller parasitics and a larger g_m-reduction ratio.*

Another circuit trick that is commonly used to reduce sensitivity to internal parasitic time constants is to include a transconductance reduction ratio in the output stage of the amplifier. Such a technique is shown implicitly in the layout example of a wide-range amplifier in Mead's text ([1], color plate 7), and is used to a rather extreme degree in Steyaert's circuit [33]. This technique is also mentioned in Sansen's tutorial [31] as a method for improving the phase margin of the amplifier.

Figure 3.7 shows experimental data across the stability boundary. As the conductance ratio increases past the stability limit, the circuit begins to oscillate spontaneously. The measured data show a smooth increase in oscillation amplitude, rather than a sudden occurrence of instability. We can attribute this smoothness to the nonlinearities of the amplifiers the circuit comprises; an increase in signal amplitude leads to changes in the average transconductances of various circuit elements. In this case, the changes serve to stabilize the oscillations at a finite amplitude.

Nonlinear analysis

Models In order to examine the behavior of the DIFF2 circuit for larger signals, we need to have some reasonable models for the circuit components. Neglecting internal parasitics, we can use equation (3.1) as a model for a transconductance amplifier. For very large signals, the $\tanh(\cdot)$ function looks rather like the $\mathrm{sgn}(\cdot)$ function. Although $\mathrm{sgn}(\cdot)$ is not linear, it is *piecewise constant*, and so is quite simple to analyze in a circuit. A more detailed analysis might involve a piecewise linear model, a Taylor expansion of the $\tanh(\cdot)$ function, or even the full $\tanh(\cdot)$ function. A note of caution: a Taylor expansion has a radius of convergence limited to the nearest singularity, and the $\tanh(x)$ function has singularities at $x = \pm i(n\pi - \frac{1}{2})$. In the final section we even consider a hybrid model in which one amplifier is considered to behave as a $\mathrm{sgn}(\cdot)$ while the other is considered to behave linearly. This hybrid approach applies when we bias the circuit to be highly resonant, so the amplitudes of the signals on nodes V_1 and V_2 are drastically different.

Very large signals — piecewise linear trajectories In the limit of very large signal amplitudes, we can use the simple piecewise constant transfer function for our amplifiers. We can easily answer the question, "when is the circuit stable for large signals?" In finding the stability limit, we are once again primarily concerned with how the circuit behaves with *constant* inputs, or "no input," because if the output goes to infinity with no input, we have no chance of finite output for nonzero input. This is an informal statement of BIBO stability (bounded-input, bounded-output). Under the piecewise-constant model, we have the circuit equations:

$$I_1 = \tilde{C}_1 \frac{dV_1}{dt} = I_{b1} \mathrm{sgn}\left(\Lambda_T \left(V_a - V_2\right)\right) \tag{3.14}$$

$$I_2 = \tilde{C}_2 \frac{dV_2}{dt} = I_{b2} \mathrm{sgn}\left(\Lambda_T \left(V_1 - V_2\right)\right) \quad . \tag{3.15}$$

These equations can be simplified, particularly if we take all voltages with reference to V_a, to the form:

$$\dot{V}_1 = -\frac{I_{b1}}{C_1} \mathrm{sgn}(V_2) \tag{3.16}$$

$$\dot{V}_2 = \frac{I_{b2}}{C_2} \mathrm{sgn}(V_1 - V_2) \quad . \tag{3.17}$$

These circuit equations give voltages at V_1 and V_2 that are piecewise-linear functions of time, so we can analyze stability by measuring polygons, in much the same way as Mead ([1], chapter 11). We can see clearly that the circuit state will follow a piecewise linear trajectory, with breaks at points where $V_2 = 0$ or $V_1 = V_2$. A stable trajectory will be a polygonal spiral into the origin of the V_1–V_2 state plane, while an unstable trajectory will be a polygonal spiral out to infinity (see Figure 3.8). The condition for stability can be expressed geometrically as a requirement that each successive pass of a state trajectory across a particular break line must be closer to the origin than the previous pass across the same break line. This condition is illustrated in Figure 3.8(b); point P' must be closer to the origin O than point P. This geometric expression leads to essentially the same sort of polygon analysis found in Mead's text. In the case of the DIFF2 circuit, we can find that the circuit is unconditionally stable for very large signals. Figure 3.9 shows experimental data for a DIFF2 filter circuit recovering from a large input step.

Moderate amplitude behavior In order to understand the behavior of the DIFF2 circuit at intermediate signal amplitudes, we must use a more elaborate model of the

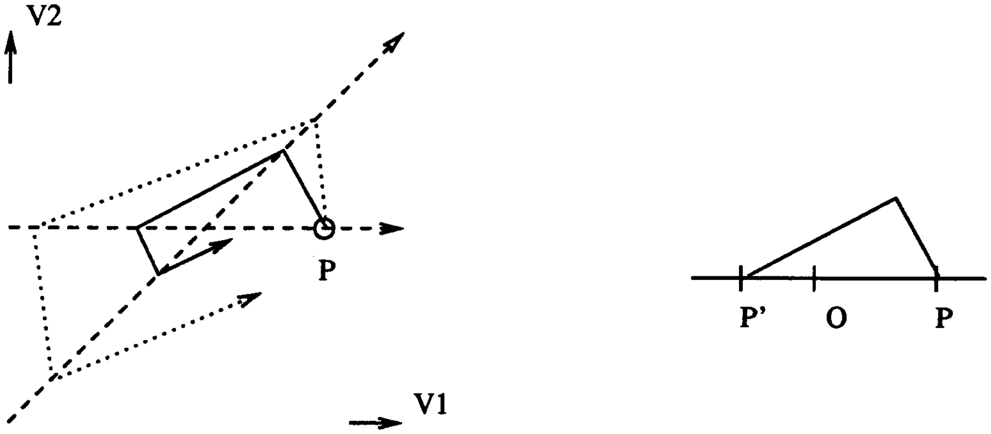


Figure 3.8: **Polygonal spiral trajectories for the piecewise-constant circuit model:** *(a) The solid line shows a stable trajectory from the initial point* P, *and the dotted line shows a hypothetical unstable trajectory from* P, *(b) The triangle which must be analyzed to determine the stability of the circuit.*

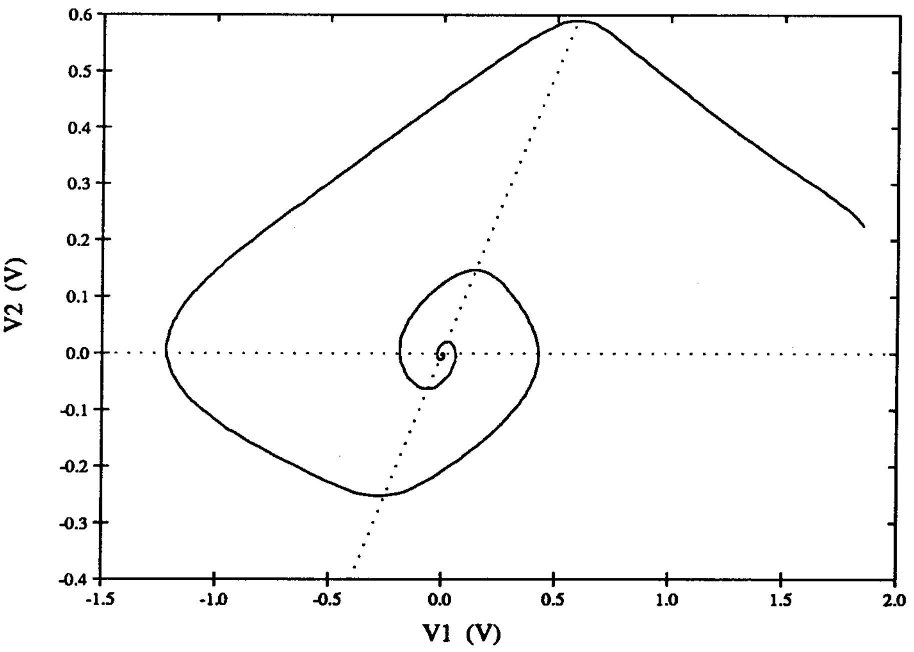


Figure 3.9: **Experimental data for the DIFF2 circuit:** *response to a large input step shows the expected polygonal shape. Note the rounding of the corners as the amplifiers pass out of the "very large signal" range*

amplifiers than that used in the previous section. What can we do with the model of equation (3.1)?

First, we can consider the average behavior of the circuit for sinusoidal inputs. As the signal on the input of one of our amplifiers grows, the state of the amplifier travels further from the origin of input–output phase space, and its average transconductance is therefore *reduced* by the compressive nonlinearity of the tanh(·) function. This reduction of average transconductance should, in turn, give a reduction in the observed resonant frequency of the circuit.

Now we have a strange thing: the resonant frequency of the circuit depends on the amplitude of the applied signal. Not only that, but, for a highly resonant circuit operating near resonance, we can have relatively large signal levels on the V_1 node even for small signals on the input. This effect can give a very strange behavior: the resonant frequency shifts downward as the *frequency* of the applied signal approaches the resonant frequency. This amounts to a bending of the normal resonant peak into a shark-fin shape. In extreme cases, we can even observe a hysteretic behavior in which there is a range of frequencies where the circuit will resonate in two distinct modes, either large-signal or small-signal, for a given input amplitude.

Figures 3.10 and 3.11 show frequency-response data for a normal DIFF2 filter and one constructed with an element with a sinh(·) characteristic. Note the bending to higher frequencies for the sinh(·) circuit, as predicted by the informal analysis above. The sinh(·) element was constructed of a circuit similar to that of Banu and Tsividis [26], but could probably be made better by a modification of Mead's horizontal resistor ([1], chapter 7; also section 5.1 of this text).

One might wish for a slightly more rigorous approach to prediction of the circuit behavior. Various methods have been developed to cope with just this sort of problem (see [29] and [30] for some examples). We can find two critical features of the circuit's frequency response in a fairly simple and straightforward way.

First, we can say that the circuit response is approximately linear below some critical amplitude V_{lst}, the "large-signal threshold." At room temperature, for the fabrication process used for the experimental circuits, V_{lst} is about 100 mV. The shaded areas in Figures 3.10 and 3.11 correspond to signal levels larger than this threshold voltage. The $\tanh(\cdot)$ function changes smoothly from linear to constant behavior, so we would expect the change in circuit behavior to follow such a smooth transition also, and, indeed, one can see that the change in behavior occurs smoothly across the shading boundary.

Next, we can find a close approximation to the "backbone" curve by considering the large-Q limit behavior of the circuit. For very large Q, $I_{b1} \gg I_{b2}$, so that V_1 takes very large excursions from zero, while V_2 remains close to zero. This lets us take the large-signal limit approximation on the second amplifier and the small-signal limit approximation on the first amplifier. These approximations give us the following circuit equations:

$$\dot{V}_1 = -\Lambda_T \frac{I_{b1}}{C_1} V_2 \tag{3.18}$$

$$\dot{V}_2 = \frac{I_{b2}}{C_2} \operatorname{sgn}\left(\Lambda_T \left(V_1 - V_2\right)\right) \quad . \tag{3.19}$$

Integration of Equations (3.18) and (3.19) gives a V_2 that is a triangular waveform with slopes of $\pm\frac{I_{b2}}{C_2}$, and hence a V_1 that is a string of parabolic sections. Integration of equation 3.18 gives us something like

$$V_1 = \beta - \frac{\omega_0^2}{2\Lambda_T} t^2 \quad , \tag{3.20}$$

where β is a constant of integration.

This solution is valid only for $V_1 > 0$, while on the other side of zero, we get a corresponding parabola with positive curvature. Beacuse the solution is parabolic, the amplitude of the solution clearly depends on the period of the waveform as

$$\beta = \frac{\pi^2 \omega_0^2}{8\Lambda_T \omega^2} + K \quad . \tag{3.21}$$

Now, for small signals, we know that the amplitude of the response peak goes to zero at $\omega = \omega_0$ because the circuit behaves linearly for small signals. We therefore choose the constant of integration K such that the backbone curve intercepts zero at ω_0:

$$\beta = \frac{\pi^2}{8\Lambda_T}\left(\frac{\omega_0^2}{\omega^2} - 1\right) \quad . \tag{3.22}$$

Figure 3.10 shows some frequency-response curves for the DIFF2 filter with the backbone curve and shading above V_{lst}. The backbone curve shown in the figure actually follows the relation

$$\tilde{\beta} = \frac{\pi^2}{8\Lambda_T}\left(\left(\frac{\omega_0}{\omega}\right)^{1.8} - 1\right) \quad . \tag{3.23}$$

It is not clear why a power law of less than 2 is needed to fit the data; it may be that the smooth transition of the tanh(·) function from "linear" to "step" behavior is responsible.

A similar analysis can be made to find the backbone curve of Figure 3.11. Because the sinh(·) function has an exponentially increasing slope as we depart from zero argument, we can assume that the *average* conductance of the sinh(·) element is close to the *maximum* incremental conductance. From an element with a current of

$$I = I_0 \sinh\left(\kappa \Lambda_T V\right) \tag{3.24}$$

this assumption about average conductance leads to a curve giving the amplitude of the resonant peak as a function of frequency:

$$A = \frac{U_T}{\kappa}\cosh^{-1}\left(\frac{\omega^2}{\omega_0^2}\right) \tag{3.25}$$

where ω_0 is the small-signal resonant frequency.

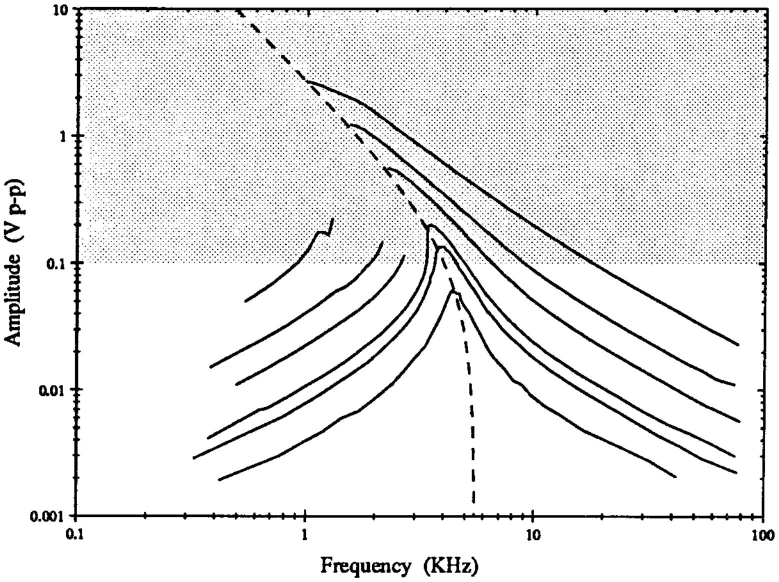


Figure 3.10: **DIFF2 filter large-signal frequency response:** *experimental data (solid) with backbone curve (dotted). The region above the large-signal threshold V_{lst} is shaded. Note the appearance of an order 1/3 subharmonic resonance peak in the largest-amplitude data.*

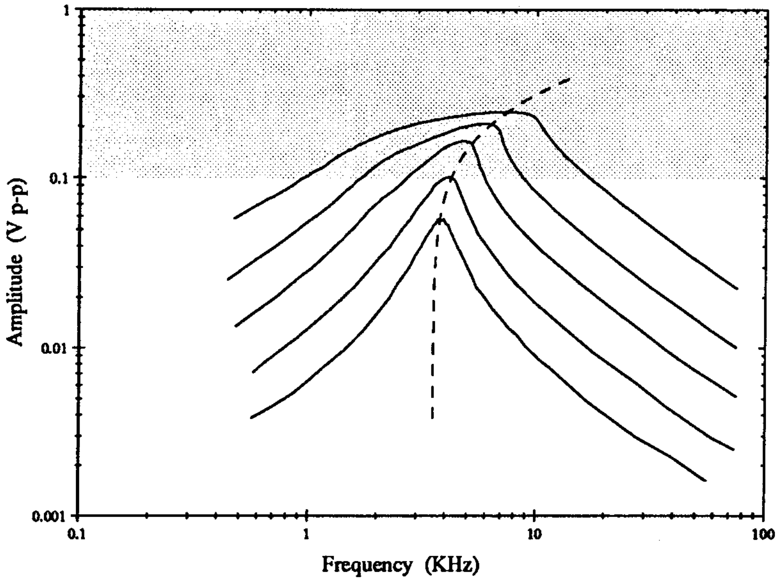


Figure 3.11: **Frequency response of a DIFF2-like filter using a sinh(·) element:** *the faster-than-linear increase of current with voltage causes an* increase *in the resonant frequency with increases in amplitude, bending the response peaks to the right. The dotted curve is a fitted backbone curve.*

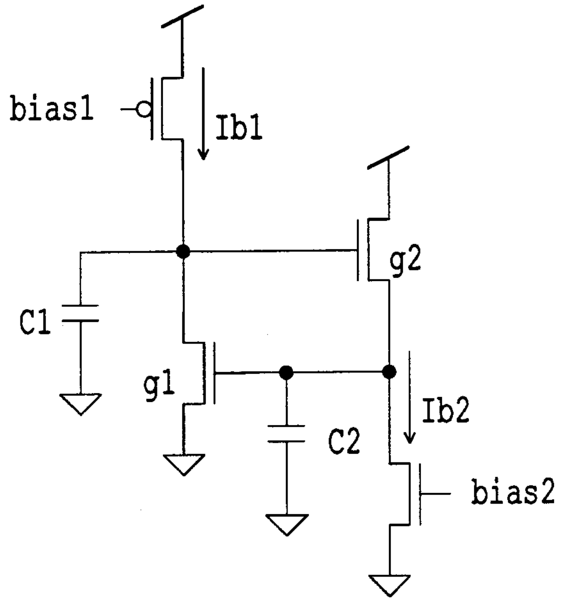


Figure 3.12: **A Four–transistor filter circuit:** ***stability problems due to internal parasitics can be largely avoided by using individual transistors as transconductors.***

The limitations due to internal parasitics can be avoided by using different circuits. For example, a very compact second–order filter circuit can be realized by using single transistors as the transconductance elements in the DIFF2 topology, as shown in Figure 3.12. Experimentally, such a filter shows approximately the same signal amplitude limit as the more complex circuit (it behaves linearly below about 80 mV p-p) without the parasitic instability.

3.2 CT delays

Time–delays are useful information processing elements. A time–delay element can be thought of as a kind of memory device, and a continuous–time delay element is thus a vector or function memory element. Such devices find uses in many areas of information processing, such as sequence recognition and synthesis, and filtering. A series of time delay elements is usually called a delay line. Delay lines are often used as time–to–space transforming devices, so that a temporal sequence is converted to a (moving) spatial pattern on the delay line.

My contribution in this section is a new way of analyzing Mead's simple first–order delay line, and several circuit extensions of the continuous–time delay line, including experimental test results from circuits I designed, had fabricated through the MOSIS service, and tested.

The fundamental description of a continuous–time delay element is

$$V_{out}(t) = V_{in}(t - \tau) \quad , \tag{3.26}$$

where V_{out} and V_{in} are the output and input signals, respectively, of the delay element. A delay line, constructed of a string of delay elements, gives samples of the original input function at successively longer delays. It is often convenient to consider the taps on a delay line as samples of a continuous wave–propagating medium which satisfies

$$\frac{\partial V}{\partial t} = -c\frac{\partial V}{\partial x} \quad , \tag{3.27}$$

where x is the position along the delay line, increasing x corresponding to increasing delays. The symbol V is chosen to represent signals because electronic signals are often (though not always) represented as voltages.

The wave–equation approach to a delay line has the flaw that it assumes a conservative medium; in fact, some wave equations can be derived from conservation laws (see, for example, Whitham's text [57]). In active electronic circuits, signal energy is *not* conserved in general, so any attempt to implement a wave–like behavior in active circuitry must be a carefully constructed approximation to such a conservative system. Parasitic passive components and second–order component behaviors will conspire to ruin many simple attempts at the construction of "conservative" circuitry.

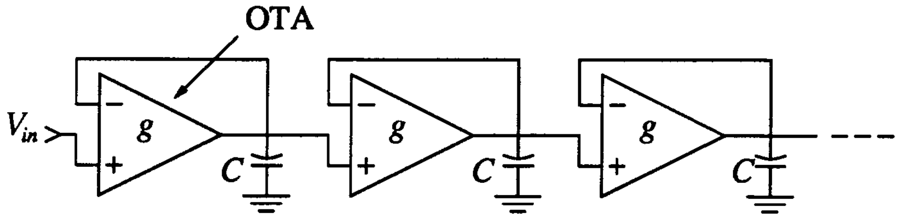


Figure 3.13: **Mead's follower–integrator delay line:** *a chain of first–order OTA–C lowpass filter sections is a simple approximation to a delay line (see [1], chapter 9).*

3.2.1 Analog delays

Analog, continuous–time delay elements ideally store a finite–length segment of a continuously variable function of time. Such a device, in principle, requires an infinite information channel capacity. What can be done with finite–capacity, practical devices?

first–order delay line A first–order lowpass filter can be considered to be a continuous–time delay element for signals of limited bandwidth. The standard frequency–domain gain and phase plots show us that the first–order filter's approximation to an ideal delay element seriously degrades at frequencies higher than the natural frequency of the filter: the gain drops significantly below unity, and the phase shift departs badly from a linear dependence on frequency.

We can, nonetheless, still obtain some delay performance from even such a simple delay element. Mead's "follower–integrator delay line" is a series of first–order lowpass sections, as illustrated in Figure 3.13 [1]. The first–order continuous–time, continuous–signal (CTCS) delay line is analyzed in the frequency domain in [1]. A time–domain analysis is also possible, and can lead to some interesting observations.

If we take the viewpoint that gave rise to Equation (3.27), we can assume that the voltages at successive stages of the first–order CTCS delay line are samples of a continuous function of both time and space. The description of each filter element is

$$\frac{dV_{out}}{dt} = \left(\frac{g}{C}\right)(V_{in} - V_{out}) \quad . \tag{3.28}$$

Now, V_{out} can be considered to be a sample of $V(x,t)$ at some position, while V_{in} can be considered to be a sample of $V(x,t)$ at some other position. According to the convention of equation (3.27), we can write $V_{out} \to V(x,t)$ and $V_{in} \to V(x-\epsilon,t)$, where ϵ has the dimensions of length. We can then use a Taylor expansion of V in x to find

$$\frac{\partial V(x,t)}{\partial t} = -\left(\frac{\epsilon g}{C}\right)\frac{\partial V(x,t)}{\partial x} + \left(\frac{\epsilon^2 g}{2!C}\right)\frac{\partial^2 V(x,t)}{\partial x^2} - \left(\frac{\epsilon^3 g}{3!C}\right)\frac{\partial^3 V(x,t)}{\partial x^3} + \cdots \quad . \tag{3.29}$$

We can see from this form that, to leading order in ϵ, the first–order CTCS delay line indeed satisfies the wave equation (3.27), with wavespeed $c = (\frac{\epsilon g}{C})$. The conclusion is that, for sufficiently large spatial–scale features (or, equivalently, sufficiently low–frequency features) of the input signal, the first–order CTCS delay line does indeed behave as a good approximation of an ideal time–delay element.

If we further examine the behavior of the first–order CTCS delay line, we can find the effect of the leading correction term. If we transform $V(x,t)$ into a coordinate system (ξ,τ) which moves toward positive x at a rate c, then, for a medium which perfectly satisfies equation (3.27), we would find no time dependence left at all in the new coordinates, as the initial waveform would be translated without change of shape at a rate c. Because the behavior of our first–order delay line is only an approximation to equation (3.27), we expect to find some interesting time dependence left after the transformation.

The transformation is straightforward, with $\tau = t$ and $\xi = (x - ct)$. Applying the chain rule, we find

$$V_x = V_\xi \qquad \text{and} \qquad V_t = V_\tau - cV_\xi \tag{3.30}$$

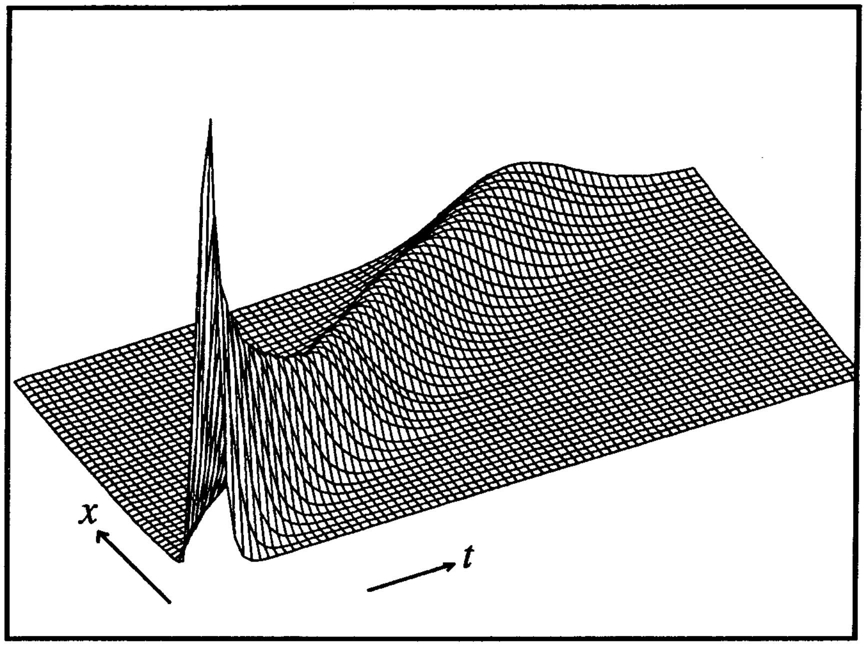


Figure 3.14: **Diffusion in first–order CTCS delay lines:** *This experimental data shows that the spreading of the initial input pulse is clearly visible as the pulse propagates.*

so that equation (3.29) is transformed to

$$V_\tau = \frac{\epsilon^2 g}{2!C} V_{\xi\xi} - \frac{\epsilon^3 g}{3!C} V_{\xi\xi\xi} + \cdots \quad . \tag{3.31}$$

Now, equation (3.31) clearly shows that, to leading order in ϵ, a waveform traveling through the first–order CTCS delay line will diffuse, with diffusion coefficient $D = \frac{\epsilon^2 g}{2C}$.

This diffusion effect is clearly visible in experiments with first–order CTCS delay lines, as shown in Figure 3.14.

We expect an impulse response of the first order delay line to look like a spreading Gaussian pulse (from the diffusion) moving along the line at a rate c. Such a function can be expressed as

$$V = V_0 A(t) exp \left[\frac{-(x-ct)^2}{\sigma^2(t)} \right] \quad , \tag{3.32}$$

where $A(t) = (t/\tau)^{-1/2}$, $c = \epsilon/\tau$, and $\sigma^2(t) = 2\epsilon^2 t/\tau$. We can rearrange terms to find a dimensionless expression of the diffusing delay:

$$V/V_0 = \theta^{-1/2} \exp\left[-\xi^2 \theta^{-1}/2 + (\xi - \theta/2) \right] \quad , \tag{3.33}$$

where $\theta = t/\tau$ and $\xi = x/\epsilon$. This kernel function fits experimental measurements to within a few percent.

correction terms through space–dependence In the spirit of numerical differentiation methods, we can take information from more than just two samples of the waveform in order to get a more precise estimate of the spatial derivative at a certain point.

For example, we could use the circuit of Figure 3.15(a) to get a second–order estimate of V_x. A linear analysis of the section behavior yields,

$$\dot{V}_i = \frac{g_1}{C}\left(V_{i-1} - V_i\right) + \frac{g_2}{C}\left(V_i - V_{i+1}\right) \quad . \tag{3.34}$$

Application of the continuum approximation and Taylor expansion technique of Equation (3.29) gives:

$$V_t = -\frac{\epsilon(g_1 + g_2)}{C}V_x + \frac{\epsilon^2(g_1 - g_2)}{2!C}V_{xx} - \frac{\epsilon^3(g_1 + g_2)}{3!C}V_{xxx} + \cdots \quad . \tag{3.35}$$

Clearly, if we set $g_1 = g_2$ then the diffusion term vanishes, leaving us with a small dispersive correction term. It is straightforward to extend this technique of adding more amplifiers to obtain corresponding correction terms to improve the spatial derivative estimation. Each additional correction amplifier must reach further from the point of estimation; this extension process can be considered as a direct analog of the reconstruction of a continuous function from a set of discrete samples. Perfect reconstruction requires a convolution of the samples with a function of infinite extent, but satisfactory results can be obtained by truncation of the reconstruction kernel.

It is interesting to note that the second–order spatial derivative estimate using feedback is very similar to Mead's second–order section. In fact, as shown in Figure 3.15 a chain of second–order sections needs only one additional transconductance ampli-

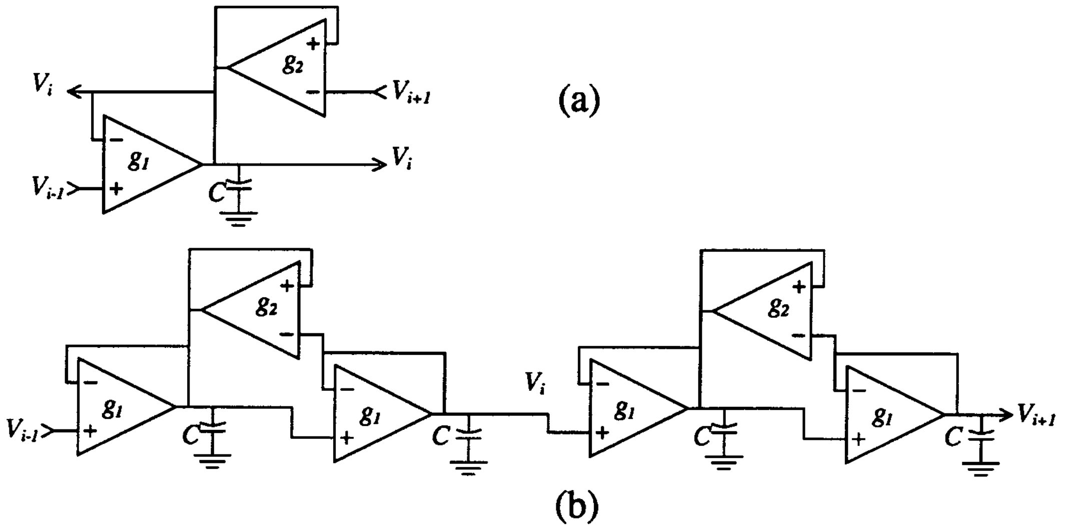


Figure 3.15: **Second–Order Spatial Derivative Estimate:** *(a) This simple implementation uses a feedback scheme much like Mead's second–order section, diagrammed in part (b) of the figure.*

fier per stage to be topologically equivalent to a chain of the sections diagrammed in Figure 3.15(a).

The chain of second–order sections has been used in cochlear models [8, 48, 11] in which a resonant behavior is required of the circuit. A simple analysis of the delay circuit of Figure 3.15(a) shows that the delay circuit becomes unstable (it develops a negative diffusion coefficient) for feedback transconductances g_2 greater than feedforward transconductances g_1. The second–order section, on the other hand, allows feedback transconductances g_2 *twice* as large as feedforward transconductances g_1 before becoming unstable. The region where $g_1 < g_2 < 2g_1$ is where the second–order section shows resonant behavior, and this region is not stably accessible with the delay circuit of Figure 3.15(a).

Positions of the poles of the circuit of Figure 3.15(a) are plotted in Figure 3.16 as the order of the circuit is increased from 1 to 11, with $g_2 = g_1$. As the order of the circuit increases, the poles asymptotically approach the imaginary axis. For an infinite–order

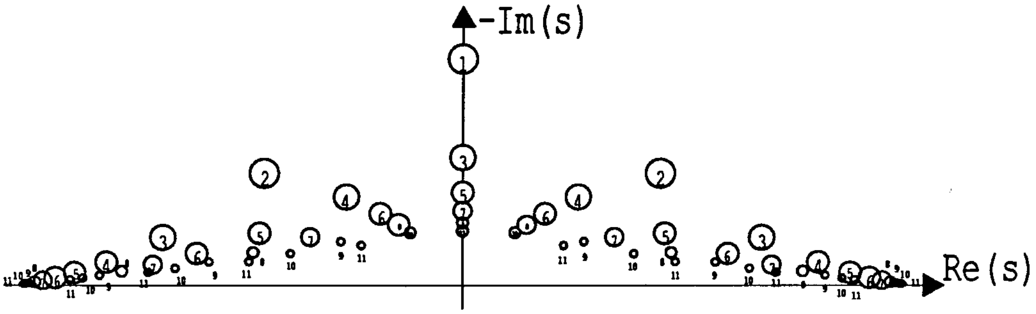


Figure 3.16: **Complex pole plot for the second–order delay line:** *Positions of the poles of the circuit of Figure 3.15 are plotted as the order of the circuit is increased from 1 to 11. The numbers indicate the order of the circuit associated with each pole. Notice the asymptotic approach to the imaginary axis.*

chain, the circuit is marginally stable, as would be expected with a diffusion coefficient of exactly zero.

After the diffusion term has been eliminated, the leading–order error term in the description of the delay chain contains a third–order spatial derivative, which gives a dispersive behavior. Dispersion (phase velocity which is not constant with respect to frequency) gradually changes a single pulse into a train of ripples. Experimentally, this dispersive effect is clearly visible in data from the delay chain of Figure 3.15(a), plotted in Figure 3.17.

Another method for cancelling the diffusive behavior of the delay chain might be to connect a circuit in what amounts to a second–order forward difference topology. By examining Taylor expansions, we can find that

$$CV_t = -\epsilon(g_1 - 2g_2)V_x + \frac{\epsilon^2}{2!}(g_1 - 4g_2)V_{xx} - \frac{\epsilon^3}{3!}(g_1 - 8g_2)V_{xxx} + \cdots \quad .$$

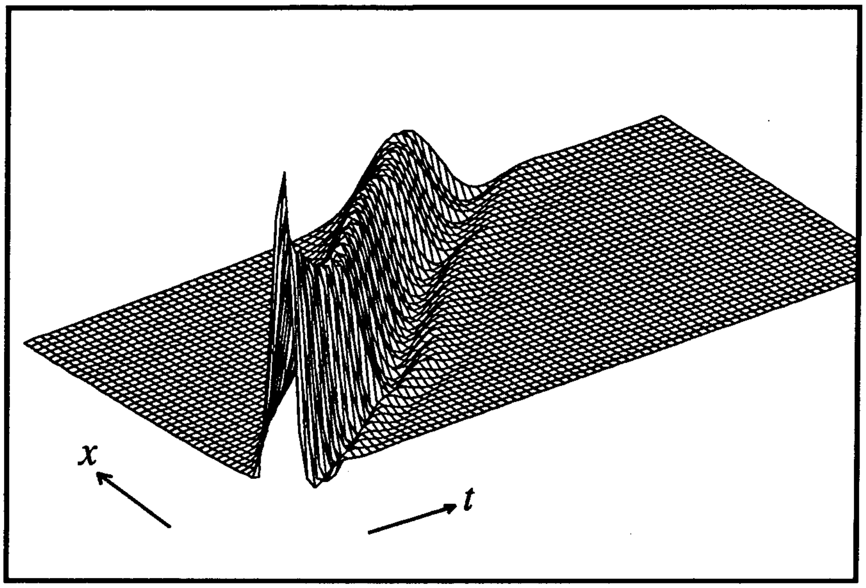


Figure 3.17: **Progress of a pulse in the delay line of Figure 3.15(a):** *This experimental data shows the gradual dispersion of the pulse into a ripple train as the leading edge progresses along the chain.*

If we take $g_1 = 4g_2$ then the diffusive term disappears, leaving us with a small dispersive term:

$$\frac{2C}{g_1}V_t = -\epsilon V_x + \frac{\epsilon^3}{3}V_{xxx} + \mathcal{O}(\epsilon^4) \quad . \tag{3.36}$$

The circuit diagrammed in Figure 3.18 is an implementation of this second–order forward difference. Experimentally, the circuit behaves somewhat differently from that of Figure 3.15(a). Note that the dispersive terms in Equations 3.35 and 3.36 are of opposite sign, indicating that the phase should be a steeper or shallower function of frequency than the group velocity. This difference between the two methods is apparent in the experimental data of Figure 3.19. A further improvement might be to combine the centered–difference and forward–difference circuits to cancel the dispersive term. Such a circuit could be built with only a moderate area cost over the existing delay sections, as one can use the shared–mirror technique of [48].

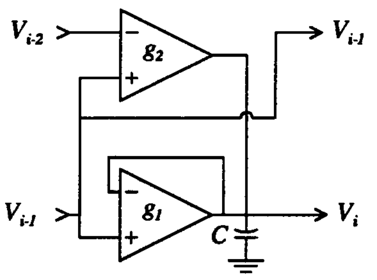


Figure 3.18: **Second–order forward difference circuit:** *The circuit can be constructed compactly using the shared-mirror technique in [48].*

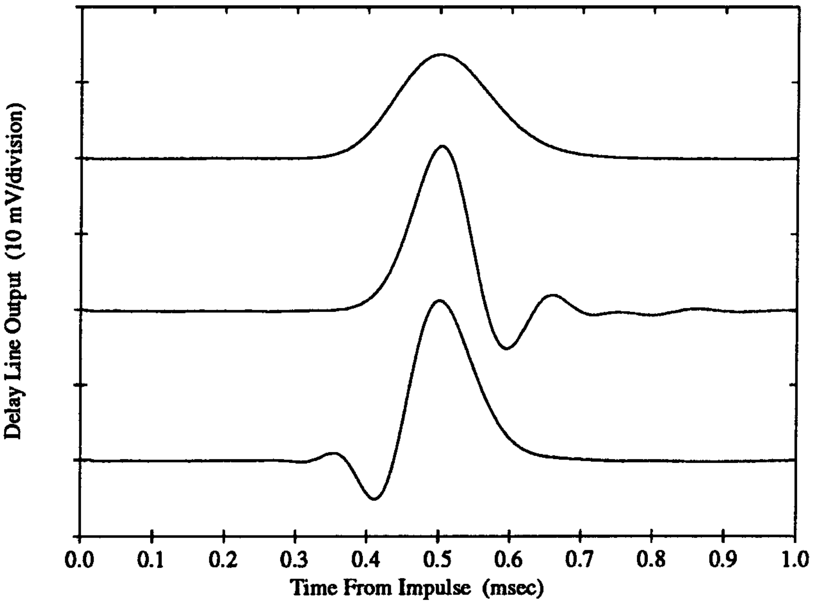


Figure 3.19: **Impulse responses of different CTCS delay lines:** *(top) first-order delay line circuit of Figure 3.13, (middle) second-order centered-difference circuit of Figure 3.15, and (bottom) second-order forward-difference circuit of Figure 3.18*

correction through time-dependence The use of filter sections with first–order space dependence (a single input and a single output), and with higher–order time–dependence is a fairly well–known technique. The Bessel–Thompson family of filters provide a phase response that is highly linear with respect to frequency over the passband. A linear phase characteristic amounts to a pure delay. An attempt to examine the Bessel–Thompson filters in the time domain using the Taylor expansion technique outlined above is not illuminating.

quantization error In all of the various schemes discussed above, the continuum approximation depends on the features of the waveform being unable to "see" the discrete nature of the medium. The fact that the medium is, in fact, discrete is expressed by the correction terms of order ϵ that are always following us around. This error is an inexorable expression of the fundamental difference between a continuous medium and a discrete medium. In the case where we actually *want* a discrete medium, as in a simulation of a mass–and–spring chain, there is no granularity error because the system to be simulated is intrinsically discretized, and there is no need to make a spatially quantized aproximation. The work of Watts [11] points the way to robust implementations of bidirectional chain simulations, such as a mass–and–spring chain. Naive attempts to build such chains using OTA–C techniques have met with disaster in the form of uncontrolled offset accumulations, preventing collection of experimental data of any significance.

3.2.2 Digital delays

Discrete signals are often advantageous in a computing system, as they can be made immune to noise by restoration processes, unlike continuous signals. It is easy to build continuous–time, discrete–signal (CTDS) delay elements. Such delays can be used in a variety of ways: the delay can be used directly as part of a computation, the delay can be used in oscillators and clock generation, or, one can build CTDS relatives of the filter–cascade cochlea for signal processing.

In Carroll's work, CTDS delay elements measured the cost of wavefront propagation in a particular direction through an array. This wavefront–processor was used to solve minimum–cost routing problems through a planar graph (*e.g.*, a printed–circuit board) [56].

CTDS delays in the form of gate delays are routinely used to design clock–generation circuits of various types. Extension of the gate delay concept by introducing a current limit in the gate output allows a simple means of obtaining adjustable delays. Adjustable delay elements of the type diagrammed in Figure 3.20 can be tuned over an enormous range of time scales, simply by changing the current limit level. In fact, the measurement of the delay period of a CTDS delay element provides a much more sensitive measurement of the device currents than direct connection of an external ammeter. Currents in the range of 10^{-14} A can be easily measured, simply by counting a time delay. Figure 3.21 plots transistor current measured by this indirect method against the gate bias voltage. Similar methods for measuring currents in the range of 10^{-17} A are described in Chapter 4.

CTDS delay elements strung in a long chain, with an exponentially increasing delay, form a relative to the filter–cascade cochlea model of Lyon and Mead [8]. The CTDS elements actually perform an operation more like median filtering than lowpass filtering, but there is a strong similarity between the two filter types and their associated signal processing operations. The CTDS chain has the advantage of having a single bias connection for each element, corresponding to delay adjustment, and no stability problems, as there are no feedback loops.

Pulse width discrimination is an example of a task for which the CTDS chain is

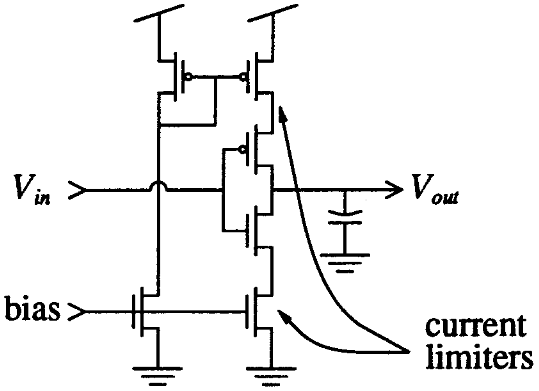


Figure 3.20: **A CTDS delay circuit:** *Adjustable delay is obtained by limiting the current available for switching state of the output.*

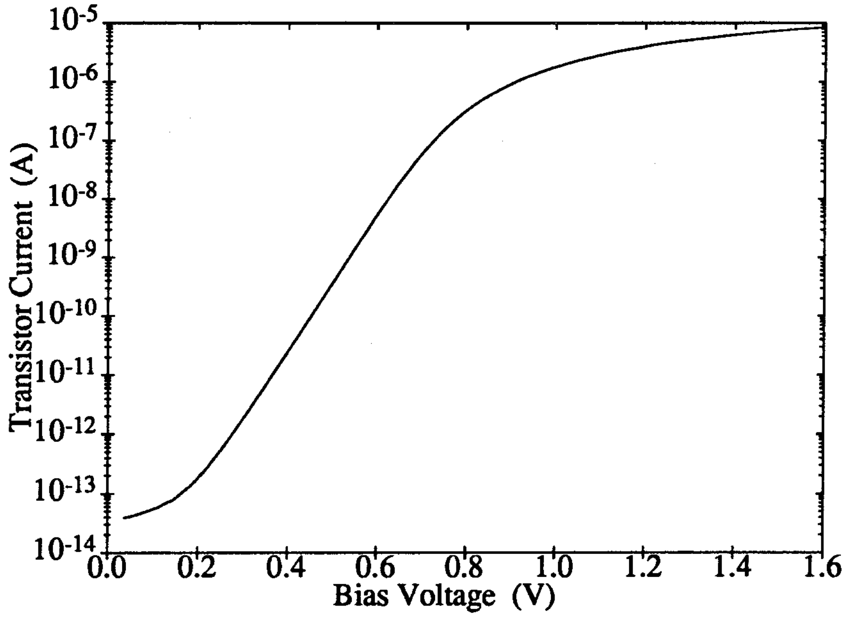


Figure 3.21: **Using a ring oscillator to measure small currents:** *Indirect measurement of the bias current of a CTDS delay element by measuring its delay period provides a very sensitive current measurement technique.*

well–suited. The CTDS chain is a tunable extension of the discriminator presented by Rahkonen [61].

Chapter 4

UV Floating–Gate Techniques

Floating–gate MOS devices have been in common commercial use for several decades in digital EPROMs, but it is only relatively recently that floating–gate charge storage has been proposed for use in analog circuitry [49, 34, 53].

UV photoinjection has been used for a long time as the means of erasing EPROM devices, but, again, only relatively recently has this effect been proposed as a means of actively *writing* charge to a floating–gate MOS device [35]. Very recently, Mead has proposed the use of UV photoinjection in active analog circuitry to compensate offsets and other device mismatch effects [3].

My contributions in this area have been the experimental characterization of simple UV photoinjection circuits, and the development of UV photoinjection circuit techniques suited to a no–frills CMOS process, specifically, $2\,\mu$m double–poly CMOS available through the MOSIS service during the late 1980s and early 1990s.

4.1 UV photoinjection device characteristics

4.1.1 A simple physical model

The usual simplified semiconductor band theory allows us to construct a simple model for the UV photoinjection (UVPI) device. Early work on Si–SiO$_2$ interface properties

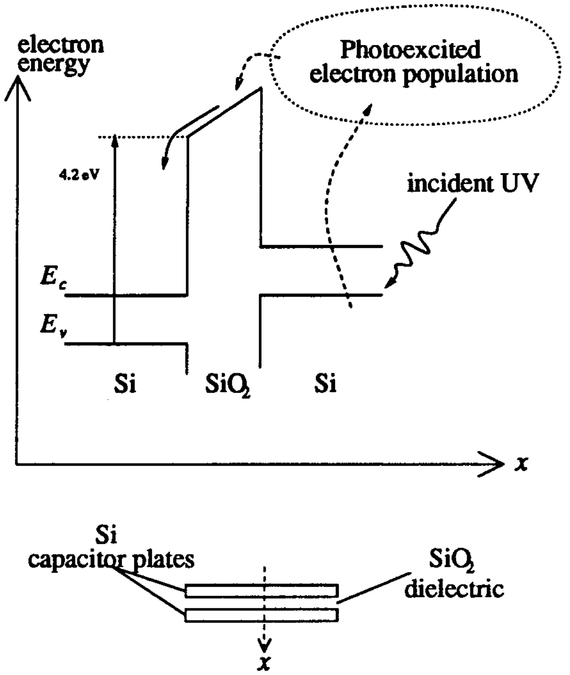


Figure 4.1: **Band diagram of UVPI device:** *UV photons excite a population of electrons from the Si to energies sufficient to traverse the Si–SiO_2 barrier and support a current. x is the direction normal to the wafer surface as the device is typically constructed in an IC process.*

showed a barrier energy of 3 eV–4 eV from Si to SiO_2 [37]. Easily available shortwave UV sources (low–pressure mercury vapor arc tubes) emit primarily in the range of 4.8 eV (254 nm), more than enough energy to excite electrons into the SiO_2 conduction band. A Si–SiO_2 interface becomes an electrical contact to the bulk oxide under UV irradiation, owing to the population of electrons excited by UV photons to energies greater than the Si–SiO_2 barrier height.

After electrons enter the bulk SiO_2, they will travel up the electric field gradient, as in any electronically conducting bulk material. A practical UVPI device usually has at least two contacts, so that elecrons injected from one contact may be collected at another. In the case where the contacts are constructed of differently doped silicon, the condition for zero net UVPI device current is *not* the same as thermal electronic equilibrium, hence, the UVPI device will reach zero net current at a nonzero applied voltage. One would expect the zero–current voltage of a UVPI device to be proportional to the work–function difference between the contact materials. A circuit model for the UVPI device might look like a voltage source in series with an impedance.

Typical UVPI device current–voltage characteristics are difficult to measure directly because of the very small currents involved. Good estimates of the I–V characteristics can be obtained by indirect means, however.

4.1.2 Experimental methods

One of the most straightforward methods for determining UVPI device current is measurement of capacitor charging rate. A circuit node with a known capacitance to constant voltage will charge at a rate directly proportional to the charging current:

$$q = CV \quad \Rightarrow \quad I = C\frac{dV}{dt} \quad \Rightarrow \quad \frac{dV}{dt} = I/C \quad . \tag{4.1}$$

If the floating node is connected to the input gate of a voltage follower circuit, then it is a straightforward experiment to measure both the voltage across the UVPI device and the rate of change of voltage on the capacitor node. From this information, we can use Equation (4.1) to calculate the device current.

4.1.3 Measured device characteristics

Experimental measurements on UVPI devices show that such a device acts, to first order, as a simple conductance in parallel with the capacitance intrinsic to the structural geometry. For example, a poly–poly UVPI structure has the I–V characteristics shown in Figures 4.2(a) and 4.2(b). (The difference between the two is an exchange of poly layers in the circuit). The shape of the I–V curve is nearly linear over a wide range of voltages. At low voltages, there is some flattening of the curve, possibly due to trapping mechanisms [54, 55], and there is some asymmetry between the first–quadrant

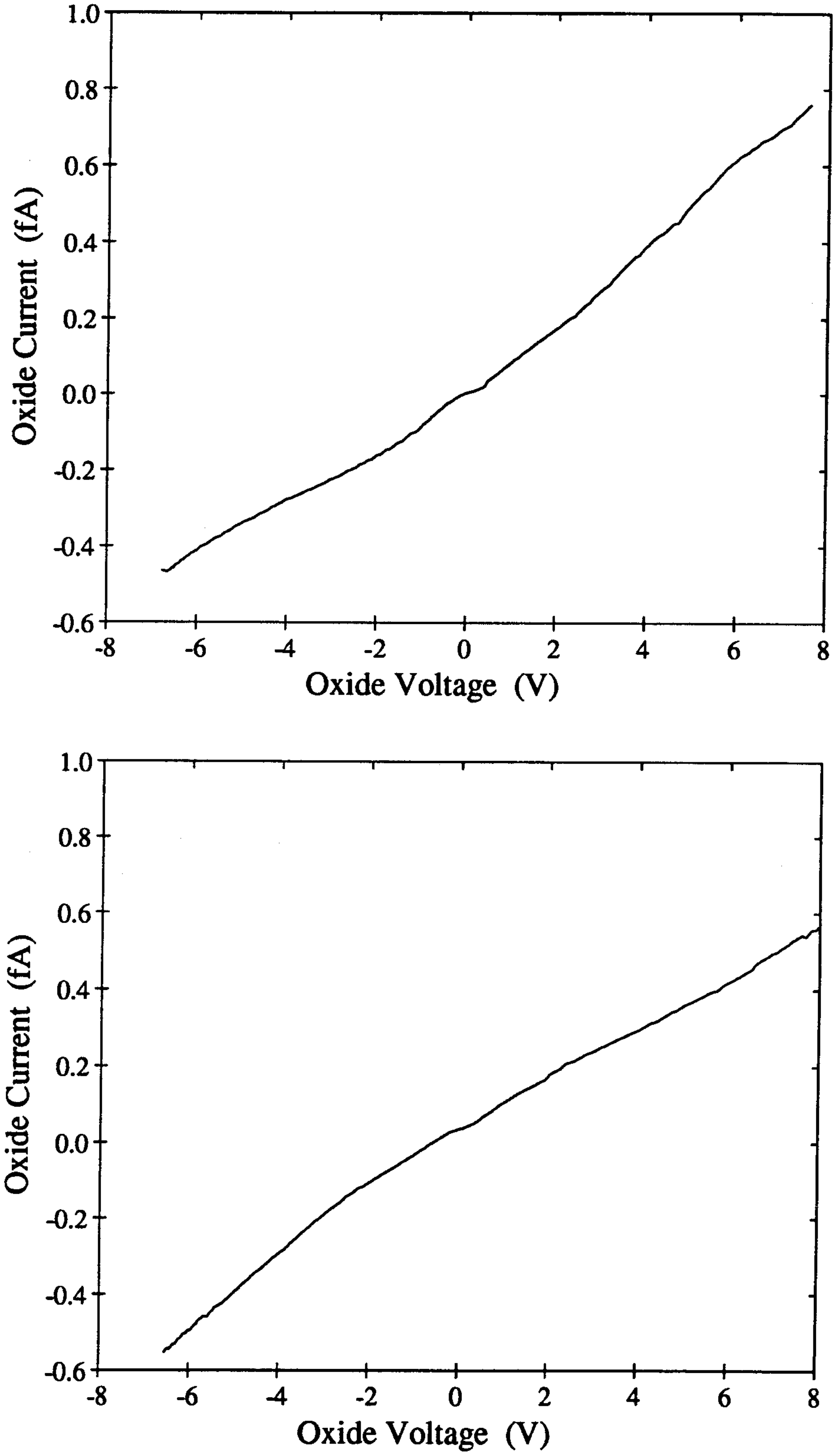


Figure 4.2: **Conduction between poly layers:** *(a) poly2 floating, (b) poly1 floating*

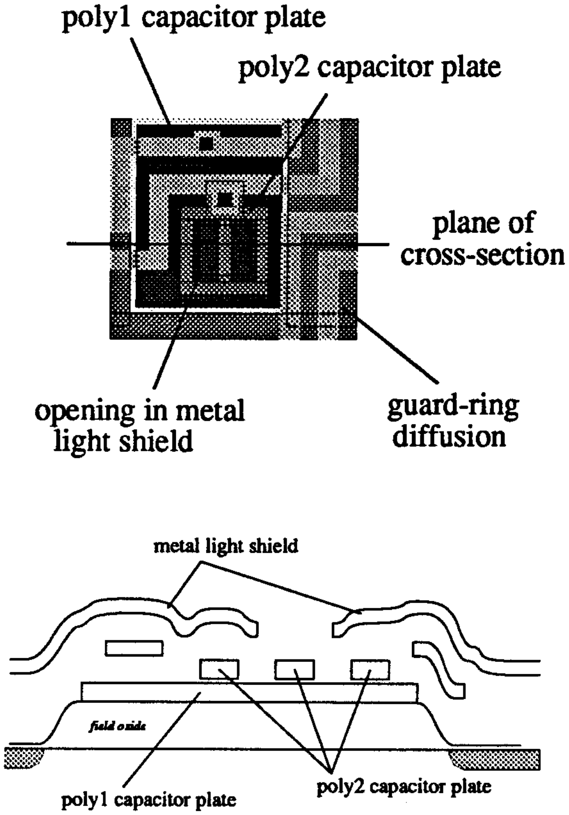


Figure 4.3: **Layout and cross–section of a UVPI device:** *the second metal layer in the fab process is dedicated to shielding most of the circuitry from incoming UV light. The relative thicknesses of the layers are approximately to scale, showing the reason for the concentration of activity at the drawn edges of the polysilicon layers.*

and third–quadrant portions of the curve, probably due to the geometric asymmetry of the UVPI structure.

Figure 4.3 shows both layout and cross–section views of an example UVPI structure. The structure of Figure 4.3 is that used in the UV detector of section 4.2, and is substantially similar to that used to gather the data in Figure 4.2.

UVPI devices show a conductance that is proportional to both the active area of the device and the intensity of the UV light illuminating the structure. Figure 4.4 plots the observed UVPI conductance for a set of test structures which are identical except for a stretching of one dimension, increasing the active area available for photoinjection.

Figure 4.5 plots the observed UVPI conductance of a single test structure exposed to different levels of illumination.

The data in Figure 4.4 clearly indicate a linear dependence of conductance on the changing device dimension. The active region of a UVPI device is concentrated at the edges of the upper polysilicon layer. Silicon absorbs UV very strongly, and the polysilicon layer in a generic CMOS fabication process is thick enough that none of the UV light is permitted through the layer. The oxide thickness between the silicon layers is typically 60 nm, far less than a UV wavelength, so light propagation between the silicon layers is only allowed in the TE mode. We might reasonably expect only a tiny fraction of the incoming UV energy to be scattered between the plates with the proper polarization. The only regions in which both Si–SiO_2 interfaces in the three–layered structure of Figure 4.1 are exposed to a substantial UV energy flux are at the edges of the uppermost Si layers. Further experiments using identical–area UVPI devices with varying perimeter length verify that the UVPI device activity can be considered as a per–unit–length effect [42].

The data in Figure 4.5 indicate that UVPI device conductance varies with illumination intensity. The variation follows a power law of 0.93, just short of linear. It is not clear at this time whether the conductance is truly nonlinear or whether the equipment available for measuring the UV intensity was inadequate for the task. Intensity at the test circuit was varied by changing the distance from a low–pressure mercury arc tube to the test circuit. The same set of physical positions was used for an intensity measurement (with a Si–diode photometer probe at the test location) and for the experimental measurement (with the UVPI circuit at the test location). The relative intensity was measured using a silicon–diode photometer intended for the visible and near–IR ranges, and calibration points for absolute intensity were taken from the UV

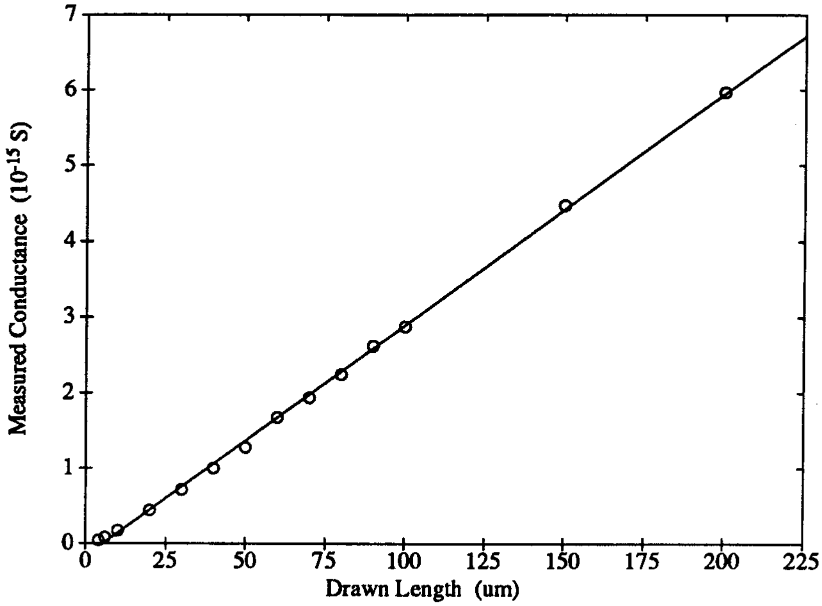


Figure 4.4: **UVPI conductance** *vs* **device size:** *UVPI device conductance varies linearly with the length of the exposed Si edges in the device. The nonzero intercept point is probably due to lithographic deviations from drawn device dimensions.*

lamp manufacturer's specifications.

UVPI devices can also be formed between the bulk silicon and polysilicon layers, through both gate oxide and field oxide layers. The conductance observed through field oxide is somewhat less than that observed through a thinner oxide, as might be expected from the greater oxide thickness. Observed conductances are about a factor of four less than those for slimilarly sized thin–oxide devices. This decrease is not in proportion to the increase in oxide thickness, however; the field oxide is a factor of ten thicker than the thin oxide layers. This discrepancy might be explained in various ways. The greater oxide thickness should permit some UV light leakage laterally between the silicon layers in the thick–oxide devices, increasing the active device area. In addition, the Si–SiO_2 interfaces of the field oxide are not necessarily grown with the same care as the thinner oxide layers, which are intended for MOSFET gates. Shallow interface traps might serve as boost states for electrons, increasing the contact efficiency of the thick–oxide devices.

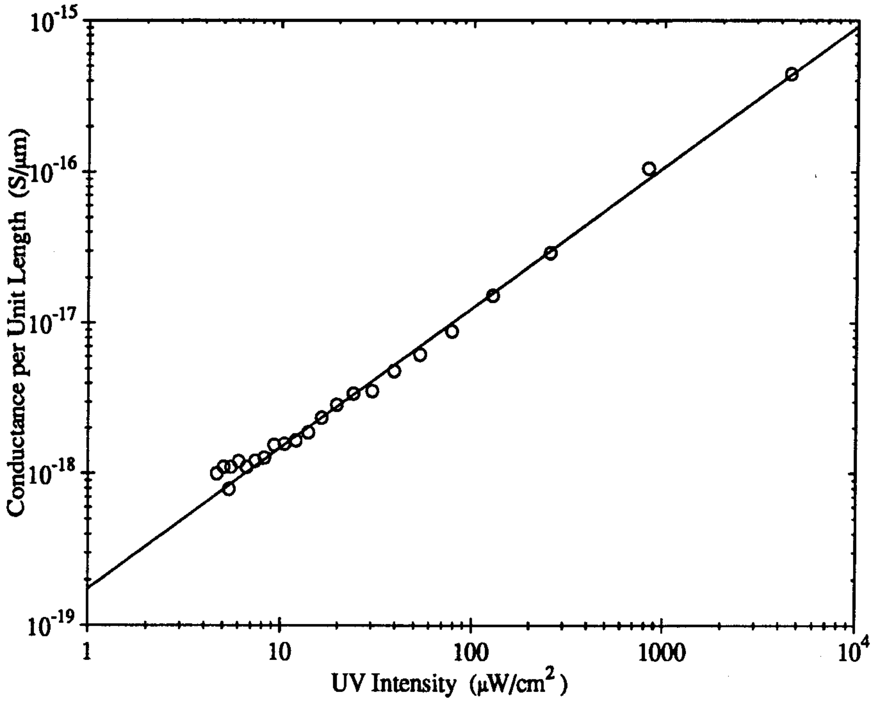


Figure 4.5: **UVPI conductance** *vs* **illumination:** *UVPI device conductance varies approximately linearly with the intensity of the UV illumination. All data were taken using 254 nm low-pressure mercury arc sources.*

UVPI devices constructed of (p-Si)–SiO_2–(n-Si) show a photovoltaic behavior analogous to a p–n junction. Measurements from a UVPI device formed between n-polysilicon and a p-diffusion in bulk silicon are plotted in Figure 4.6. The current through the device goes to zero at a voltage of about 0.8 V between the p-Si side and the n-Si side of the device, corresponding to the work–function mismatch between the p-Si and n-Si. A simple band diagram of this work–function mismatch effect is illustrated in Figure 4.7.

4.1.4 UVPI device circuit model

UVPI devices have rather simple first–order models. The geometric capacitance of the structure can be lumped into a single capacitor, the UV–enabled conduction process through the oxide layer can be lumped into a switched conductance, and any work–function mismatch between contacts can be lumped into a voltage source. Figure 4.8 diagrams this circuit model.

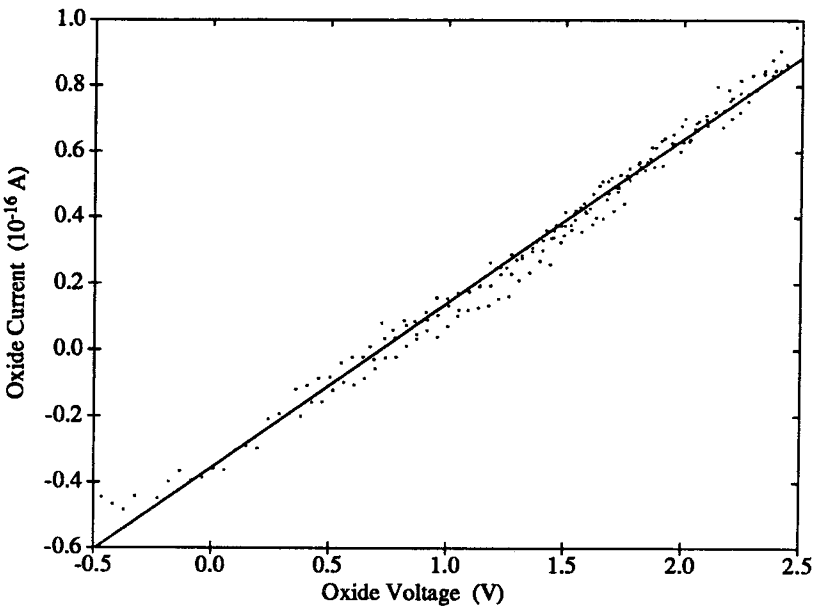


Figure 4.6: **UVPI device with p-Si–n-Si work function mismatch:** *The work-function mismatch between the p-Si and the n-Si induces an offset voltage. The voltage across the UVPI device is negative at zero current.*

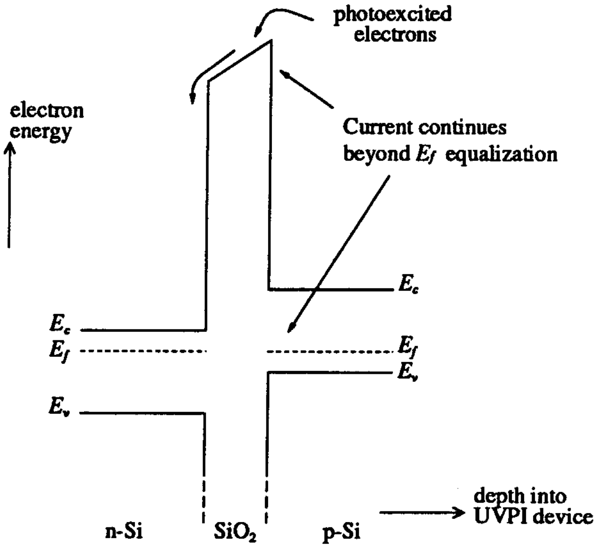


Figure 4.7: **Illustration of p-Si–SiO$_2$–n-Si work function mismatch:** *The work-function mismatch induces an electric field across the oxide. Under UV exposure, the oxide field induces a current flow even at zero voltage. This is identical to the photo-voltaic effects found in p–n junction diodes.*

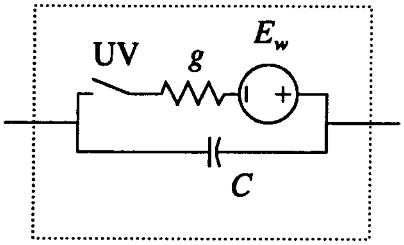


Figure 4.8: **UVPI circuit model:** *a simple switched linear circuit model is sufficient for many designs using the UVPI device*

The simple first–order model does not incorporate the subtle nonlinearities of the UVPI device I–V curve, but the model is sufficient for many circuit designs incorporating these devices. A typical use of the UVPI device is in a feedback loop with a large loop gain; such an arrangement tends to hide the nonlinearities of the devices.

There is a variety of circuit applications for UVPI devices. Past applications were for erasure and, more recently, writing of digital EPROM data [35]. More recent applications include offset nulling [3, 46, 38] and more general operating point and parameter storage [53, 38, 47, 42], as well as detection and measurement of UV light [45]. The simplest applications of the devices treat them as switches, allowing the application of switched–capacitor circuit techniques for offset nulling and parameter storage. More complex applications may involve the use of the long time constants involved in changing node charges [42].

4.2 UV detector / dosimeter

A UVPI device is a structure whose electrical properties change in the presence of UV radiation. This fact immediately leads to the idea of building a circuit to register the presence of UV light. One way to build such a circuit is to build a relaxation oscillator with a time constant that depends on the UVPI conductance. In this way, the oscillator will run when exposed to UV, and stop otherwise.

The band structure of the Si–SiO_2–Si devices is such that they are primarily sensitive to photons of energies greater than 4.2 eV [37], corresponding to wavelengths shorter than 295 nm. Peak biological activity (*e.g.*, erythema, DNA damage) is in the band from 280 nm to 320 nm, peaking at 297 nm [39, 40]. The Si–SiO_2–Si detector may therefore be adequate for a low–cost and easily interfaced UV detector for biological or, possibly, consumer applications.

It is interesting to note that the physical structure of the UV detector is virtually identical to that of MOS–transistor detectors for ionizing radiation, which use the induced space–charge in the SiO_2 layer as a dose measurement [41]. The surrounding readout circuitry tends to be different in the two cases, however.

This section will begin with a description of the UV detection structure and circuits employing the structure, and proceed to present test results from fabricated test structures. The test structures were fabricated in 2 μm CMOS bulk processes.

4.2.1 Detector structures and circuits

The basic detector structures are capacitors with doped silicon plates and SiO_2 dielectric. Transistors in a self–aligned silicon–gate MOS process are such structures, as are poly–poly capacitors (both capacitor plates formed of doped polycrystalline Si) in a double–poly (two layers of polycrystalline Si) MOS process. The poly–poly capacitor structures were the primary type used in the circuits described in this section. Figure 4.3 shows the layout and cross–section of the detector structure.

Detector Structure Layout UV–sensitive structures have quite a simple layout. Any silicon capacitor structure exposed to UV radiation will do the job, although certain geometric considerations can improve performance. In particular, it pays to remember that the edges of the capacitor are the active regions for UV–induced oxide conduction,

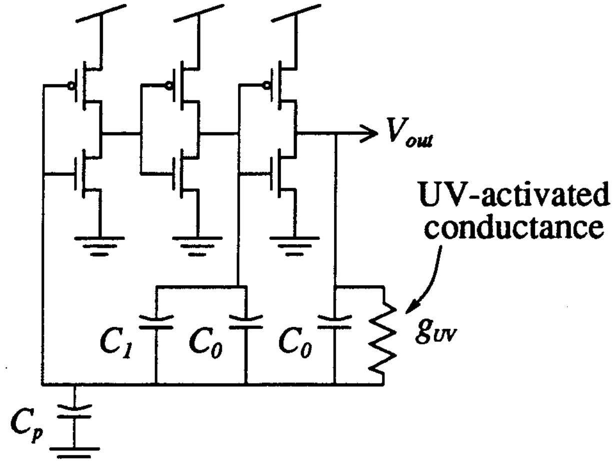


Figure 4.9: **An integrated UV detector:** *This detector is a relaxation oscillator based on Mead's pulse-firing "neuron" circuit (see [1], chapter 12).*

so the edge–to–area ratio should be large. This edge–length effect is due to the fact that the metal and polysilicon layers are opaque to UV, so the resulting shadow pattern on deeper layers determines which portions of the silicon structure are exposed to the UV radiation [42]. In general, one needs to control the geometry of the detection circuit rather carefully to prevent undesired parasitic conductances from being activated along with the intentional detector conductance. A simple technique for light–shielding is to use a metal layer to cover the portions of the circuit which should be shadowed.

Integrated Detector A simple integrated detector was fabricated in a $2\,\mu$m CMOS bulk process in an area of $63 \times 80\,\mu$m. The detection circuit used a relaxation oscillator of the type used by Mead and others as a simple model of a pulse–firing neuron (see [1], chapter 12). Figure 4.9 depicts a diagram of the circuit.

Assuming infinite–gain, symmetrical amplifiers, this circuit is easy to analyze, as under the above assumption it is a switched *RC* network. The amplifier outputs are either at V_{dd} or at ground, depending on the state of the corresponding inputs, and the rest of the circuit behaves linearly. Using such a piecewise–linear approach, the

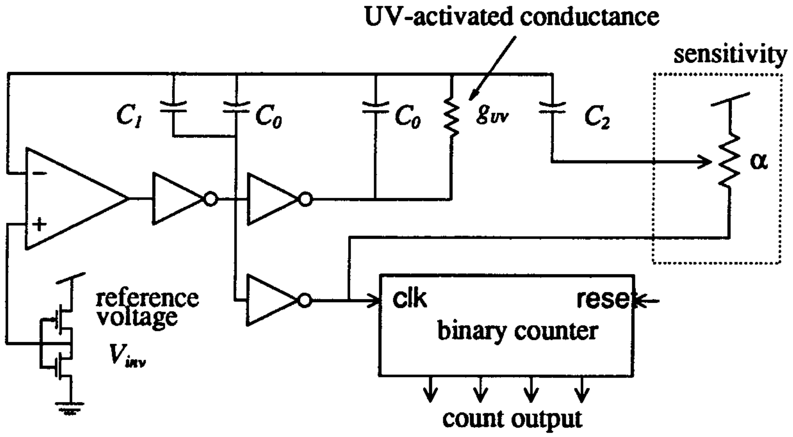


Figure 4.10: **A monolithic UV dosimeter:** *a relaxation oscillator similar in principle to that of Figure 4.9 is combined with a digital counter to register total UV energy dosage. Connection of an off–chip potentiometer as shown in the dotted box can improve sensitivity.*

oscillation period can be shown to be

$$T = 2\frac{2C_0 + C_1 + C_p}{g_{UV}}\ln(1 + \frac{2C_1}{2C_0 + C_1 + C_p}) = 2\frac{C_{total}}{g_{UV}}\ln(1 + 2\epsilon) \quad . \qquad (4.2)$$

The simple integrated detector of Figure 4.9 is functional, but it has a very low frequency of oscillation even at very high UV intensities. The low sensitivity is due to the fact that the capacitor ratio factor ϵ is rather large, approximately 0.3, and that the conductance g_{UV} is very small, $\mathcal{O}(10^{-15})$ S.

4.2.2 Monolithic dosimeter

An improved detector has been fabricated and tested. The improved detector includes an integrated digital counter, and occupies an area of 545×445$\,\mu$m. The detector shows considerably better sensitivity than the earlier version, and has provision for off–chip sensitivity improvement with a trimming potentiometer.

Figure 4.10 shows a circuit diagram of the dosimeter circuit. The capacitor C_2 allows a partial cancellation of the effect of C_1, for improved sensitivity. Ideally, for maximum possible sensitivity, the ratio factor $\epsilon = C_1/C_{total}$ should be comparable to

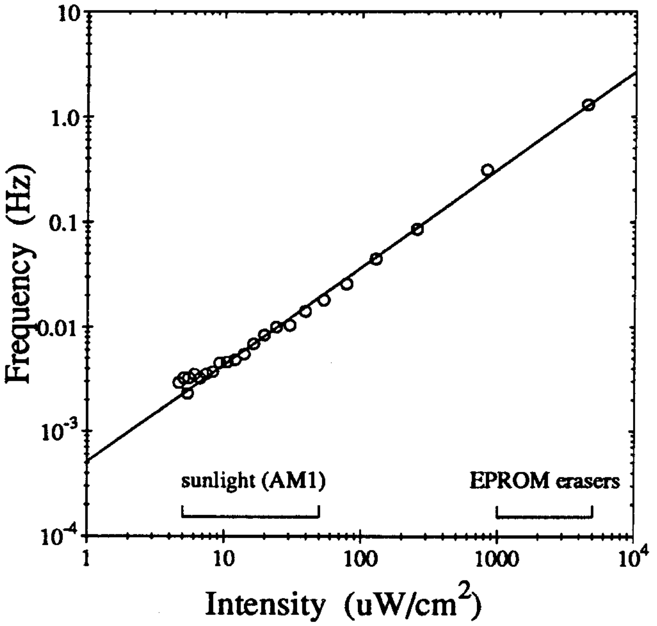


Figure 4.11: **Response of the detector circuit of Figure 4.10:** *frequency of oscillation is approximately linear in UV intensity. The sensitivity is 2.7 mJ per cm^2 per count. Typical intensity ranges corresponding to sunlight at airmass 1.0 and EPROM erasers are noted. The range given for sunlight is the typical integrated power density range of wavelengths shorter than 395 nm. Note that all data shown are for 254 nm UV.*

$1/A$, where A is the amplifier gain, typically around 60–70 dB. Such fine control of circuit parameters is not practically possible, hence the trimmable correction circuit. Experimentally, sensitivities two orders of magnitude better than the early detector are possible with the trimmed detector. Analysis of the improved detector circuit yields essentially the same result as for the simple detector:

$$T = 2\frac{2C_0 + C_1 + C_2 + C_p}{g_{UV}} \ln(1 + 2\frac{C_1}{C_{total}} - 2\alpha\frac{C_2}{C_{total}}) = 2\frac{C_{total}}{g_{UV}} \ln(1 + 2\tilde{\epsilon}) \quad , \qquad (4.3)$$

where α is the potentiometer gain factor. For the experimental results presented, $\tilde{\epsilon} = 0.013$.

Figure 4.11 plots the oscillation frequency of the detector against the UV intensity. A type G4T5 low-pressure mercury-vapor tube supplied 254 nm UV for the test results presented. The count rate varies as a power law of intensity, with a power of 0.93

— very nearly linear. Because the count rate is approximately linear in intensity, the counter registers approximately the total energy dose received by the detector. For the trim setting shown in Figure 4.11, the dosimeter sensitivity is 2.7 milliJoules per cm^2 per count. For precise applications, a state machine more sophisticated than a simple counter could compensate for the subtle nonlinearity.

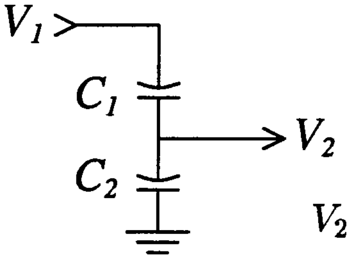
The dosimeter operates over a wide range of supply voltages, from 1.3 V to 9 V, and consumes 900 nA of supply current at 3 V. The current load varies somewhat with power–supply voltage; more sophisticated bias circuitry could give a fixed current at the expense of silicon area. Addition of the trimming potentiometer increases the current consumption to several microamperes, depending on the exact resistance value used. The low power consumption and insensitivity to supply voltage lend themselves well to inclusion in a battery–powered system.

One copy of the detector circuit has been run over seven million cycles (total UV dosage of approximately 19000 joules per cm^2, at an exposure rate of approximately 5 mW per cm^2) without any resolvable change of performance, indicating a very long lifetime.

The Si–SiO_2 ultraviolet detector can be constructed in standard MOS fabrication processes, and so allows the design of a monolithic device capable of transduction, signal conditioning, and data processing. This paper has presented an elementary example of such a device which incorporates the sensing structure, amplifier, and digital counting circuitry.

4.3 Capacitive networks

The existence of a nearly perfect insulator in silicon MOS devices allows the construction of capacitive networks which are directly analogous to resistive networks. The simple



$$V_2 = \left(\frac{C_1}{C_1+C_2}\right) V_1 + \frac{q_0}{C_1+C_2}$$

Figure 4.12: **Capacitive voltage divider:** *This is an elementary example of a linear capacitive network which is directly analogous to a linear resistive network. Capacitive networks operate on charge signals in a way analogous to resistive networks operating on current signals.*

equation describing the electrical behavior of an ideal capacitor is structurally identical to Ohm's Law, and, indeed, one can build passive circuits such as voltage dividers using purely capacitive networks.

$$\text{capacitor equation} \qquad q = CV \quad \longleftrightarrow \quad I = gV = V/R \qquad \text{Ohm's Law} \tag{4.4}$$

Straightforward application of Equation 4.4 to the capacitive circuit of Figure 4.12 leads to the following description of the circuit behavior:

$$V_1 = V_0 \frac{C_1}{C_1 + C_2} + \frac{q_0}{C_1 + C_2} \tag{4.5}$$

where q_0 is the charge on the node V_1. Thus, a capacitive network allows us to do scaling in the same way as a resistive network, but, in addition, we are allowed to *shift* voltages by setting a charge on a circuit node.

The UVPI devices provide a method for manipulating the quantity q_0 in a circuit such as that of Figure 4.12, without disrupting the excellent insulating qualities of the oxide layer (recall the long device lifetime shown in Section 4.2). One or more of the capacitances in a network can be UVPI devices which, under UV exposure, provide current paths to charge or discharge nodes in the capacitive network.

A simple application of a capacitive network in a classical circuit is the voltage

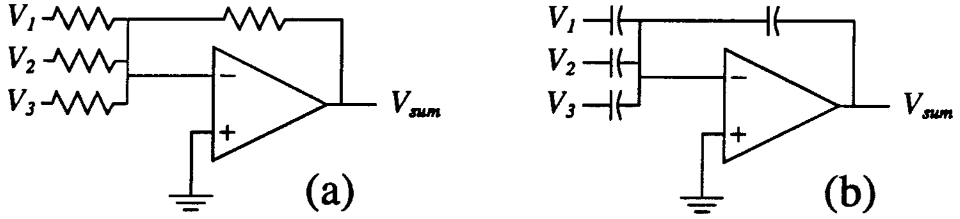


Figure 4.13: **Voltage summing amplifiers:** *(a) A classical summing amplifier design using an op–amp and a resistive network can be directly translated to (b) an analogous design using a capacitive network, suitable for CMOS technology.*

summing amplifier diagrammed in Figure 4.13(b). The capacitive network is a direct replacement for the standard resistive network shown in Figure 4.13(a).

Another application of a capacitive network in a building–block computational circuit is given in the transimpedance amplifier of Figure 4.14(b). Under UV exposure, the circuit settles to $V_{out} = V_{ref} - gI$. Thereafter, if no UV exposure is allowed, the circuit computes a current–to–voltage function $V_{out} = V_{ref} - \tilde{g}(I - I_0)$, where I_0 is the current that was applied during UV exposure, and $\tilde{g} = \frac{gC_1}{C_1+C_2}$ is the effective conductance of the feedback transconductance with the capacitive scaling network. When a typical transconductance amplifier is used for the feedback, the tanh(·) nonlinearity of the transconductance amp is inverted, giving a nonlinear transimpedance which has singular behavior for large input signals. The capacitive network alleviates this problem by scaling the voltage difference $V_{out} - V_{ref}$ that is applied to the input of the transconductance amp.

This example circuit uses the UVPI device as a switch to connect two circuit nodes transiently, then disconnects the nodes (UV off) to store the state as a charge on the floating node. In this way, the circuit's operating point is stored as a reference state for future operations; such a technique is a direct analog to switched–capacitor techniques,

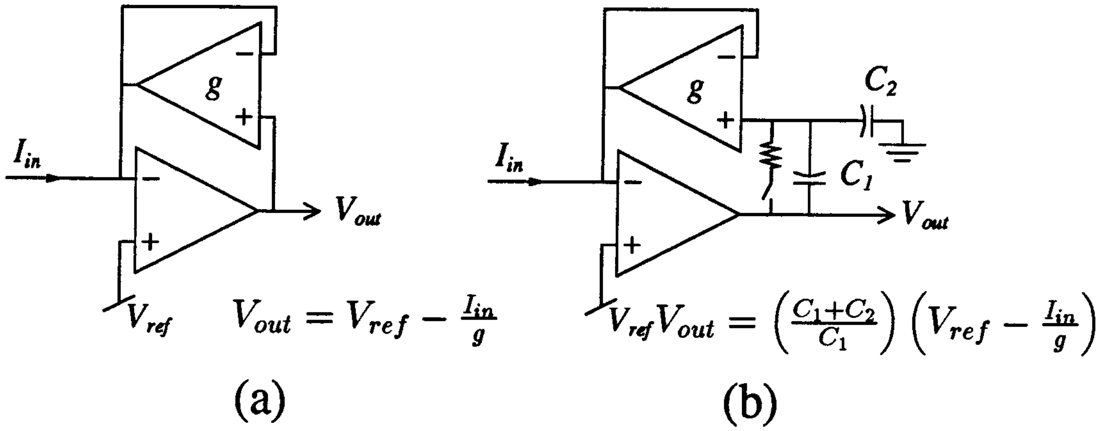

g
I_{in}
V_{out}
V_{ref}
$V_{out} = V_{ref} - \frac{I_{in}}{g}$
(a)
g
C_2
C_1
I_{in}
V_{out}
V_{ref}
$V_{out} = \left(\frac{C_1+C_2}{C_1}\right)\left(V_{ref} - \frac{I_{in}}{g}\right)$
(b)

Figure 4.14: **Transimpedance amplifiers:** *circuits suitable for integrated circuit realization use only transistors and capacitors. (a) a simple circuit using a transconductance amplifier to substitute for the resistor typically found in the discrete version of the circuit, and (b) a version including a capacitive voltage divider to linearize the transconductance.*

except that, because of the excellent insulating qualities of the oxide, the operating point is stored indefinitely, allowing DC operation.

Experimental data showing the behavior of transimpedance circuits like those of Figure 4.14 is plotted in Figure 4.15. The reduction in the effect of the nonlinearity is clearly visible.

4.4 Local offset correction

Another application of UVPI devices is for nulling of amplifier offsets. As seen in the previous section, an artfully constructed circuit can contain a feedback loop that allows a UVPI device to set the operating point. Such feedback loops can also allow the construction of amplifiers which correct their offset voltages. Such offset correction schemes using UVPI devices can be found in the work of Mead and others [3, 46, 38]. One particularly simple example is again a direct analog of a switched–capacitor offset nulling technique. The circuits diagrammed in Figure 4.16 are differential voltage amplifiers of

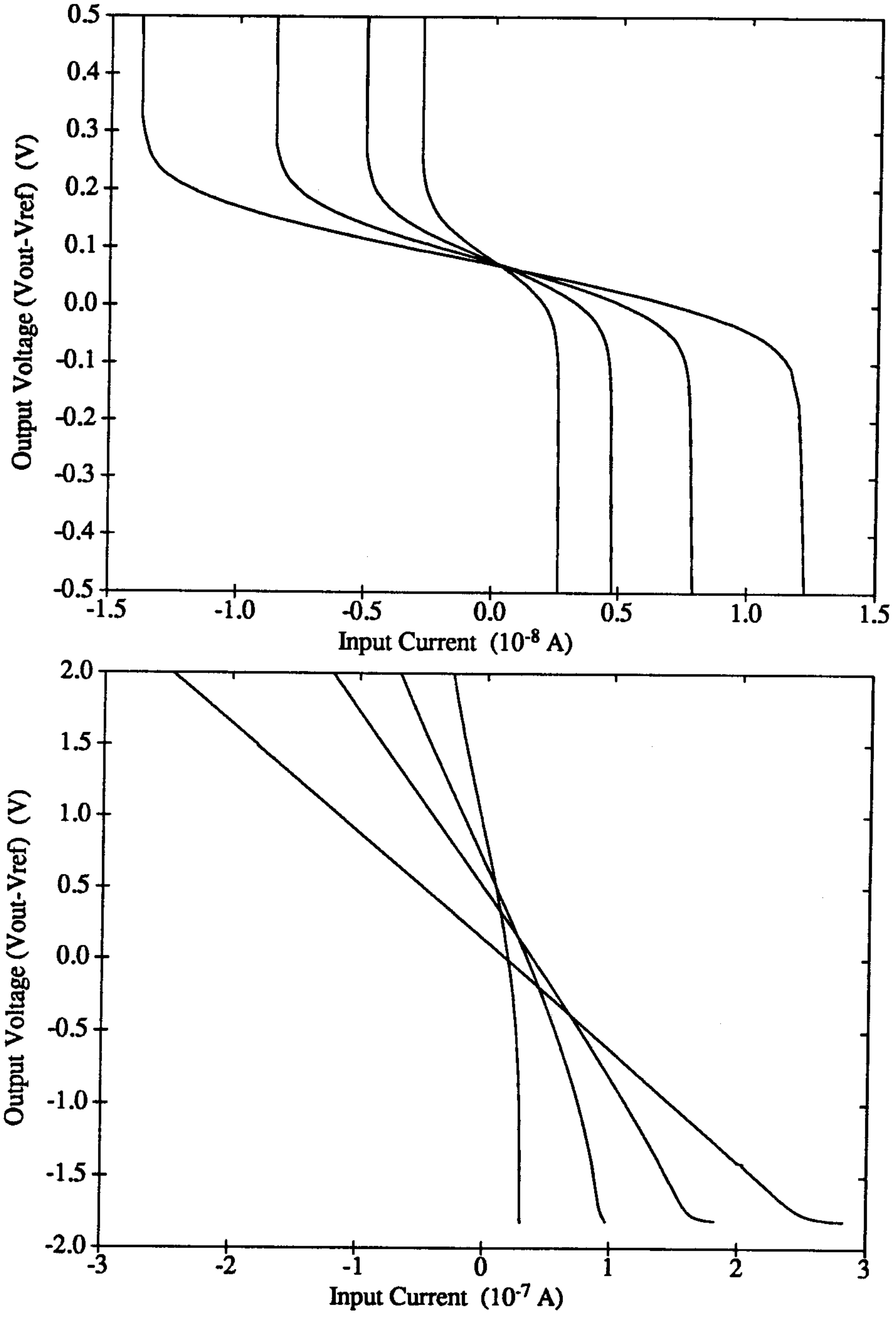


Figure 4.15: **Behavior of the transimpedance amplifiers of Figure 4.14:** *Experimental data for a 2μm CMOS realization of the integrated transimpedance circuits of Figure 4.14(a) (top) and Figure 4.14(b) (bottom) shows how the capacitive voltage divider reduces the effect of the nonlinearity of the feedback element.*

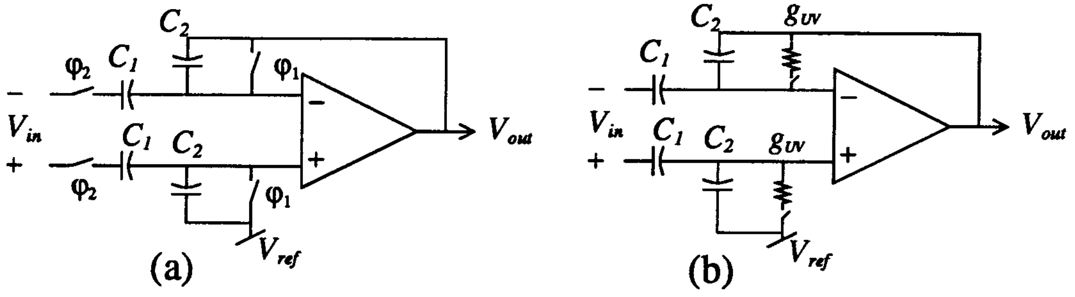


Figure 4.16: **Local offset nulling circuits:** *(a) a switched-capacitor circuit, and (b) its direct analog using UVPI devices to achieve operation down to virtually zero frequency.*

a fairly standard type, using capacitive networks around an operational amplifier to set the voltage gain. The circuit of Figure 4.16(a) is a standard switched-capacitor circuit for offset cancellation, while the circuit of Figure 4.16(b) is the UVPI implementation of the differential amplifier.

As in the previous section, the UV need only be applied once to set the operating point of the circuit. Thereafter, the setpoint is "remembered" indefinitely as the charges on the input nodes of the operational amplifier.

4.5 Global offset correction

The technique for offset correction outlined in the previous section depends both on having a large loop gain in the local circuitry, and on having a circuit configuration that lends itself to the inclusion of a capacitive network appropriate for the computation (*e.g.*, voltage scaling). It is possible to keep the philosophy of using UVPI devices as switches to enable storage of operating points, while separating the offset-correction operation from the operation of the remainder of the circuit.

For a large array of identical subcircuits to be trimmed, as in the silicon retina [4] or the silicon cochlea [48], it is possible to build a block of additional circuitry onto the

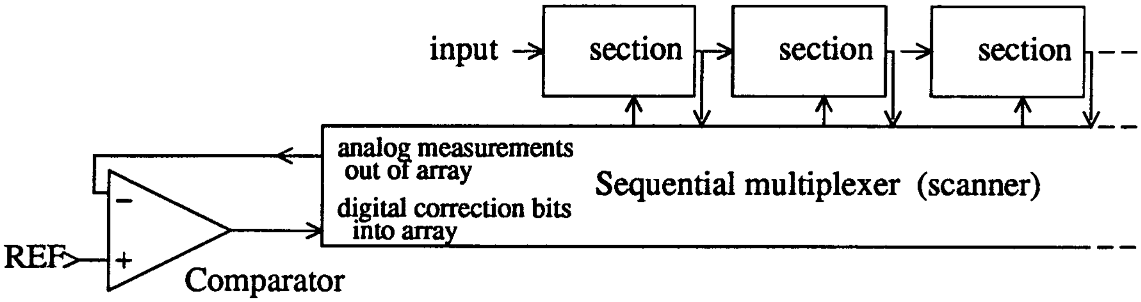

input
section
section
section
analog measurements out of array
digital correction bits into array
Sequential multiplexer (scanner)
REF
−
+
Comparator

Figure 4.17: **Global trimming circuit:** *A global trimming scheme uses a scanner to compare each unit in a large array to a single reference, and pass back a correction bit to change the offset up or down.*

silicon chip that amounts to an automated test–and–trim for each element. In this way, a very large array can be made self–trimming, reducing or eliminating the problems associated with offset voltages. The trimming circuitry can include UVPI devices as a means of storing the trim parameter for each section.

Such a global trimming technique has several advantages: (1) it uses no high voltages as in tunnelling techniques [49, 34, 36, 53] or hot–electron techniques [44]; (2) it makes use of "scanner" circuitry, which is already an integral part of many such array circuits [50]; (3) it compares each subcircuit in the array to a single reference and uses a single high–gain amplifier. Random local variations can be virtually eliminated by the combination of a single (and therefore constant) reference, and the use of a potentially enormous loop gain — because there is only one reference amplifier, separate from the array cells, there are no scaling problems associated with building a high–gain amplifier.

Figure 4.17 diagrams an example of a global–trim system. The system to be trimmed is a delay line formed of a chain of second–order sections, like that of Figure 3.15(b). One of the transconductance amplifiers in each second–order section was constructed with some additional circuitry to allow trimming of the offsets, as shown in the detailed circuit diagram of Figure 4.18. In order to ensure testability, the trimming circuitry can

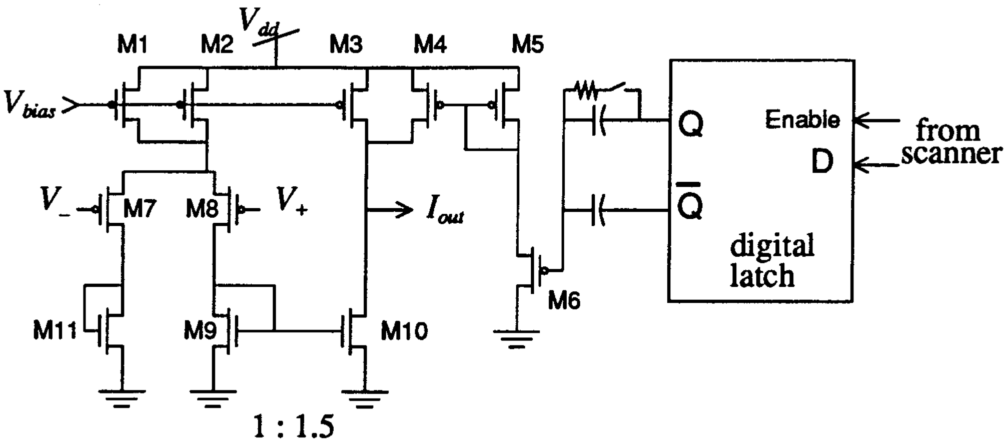


Figure 4.18: **Global trimming circuit cell:** *transistor–level diagram of the amplifier used in the global trimming scheme of Figure 4.17. The amplifier includes extra devices M4, M5, M6 in order to adjust the output offset voltage.*

only correct negative offsets. Therefore, the amplifiers were designed with a tendency toward negative offsets by the 1:1.5 mismatch in the current mirror M9–M10.

Each trimming subcircuit is composed of a one–bit static memory cell, a UVPI device with a matching capacitor, and a transistor controlled by the floating node voltage. The UVPI device is driven by the bit stored in the SRAM, while the matched capacitor is driven by the complement of the bit (both are available from the SRAM circuit), so that a change of state in the SRAM bit is capacitively coupled to the floating node with a net weight of zero (neglecting capacitor mismatch, which is a few percent at most) [51]. Meanwhile, in the presence of UV light, the UVPI device resistively couples the SRAM bit into the floating node. The UVPI device thus is charging the floating node toward either the positive or the negative power supply rail while UV light is on. The trimming process depends on the fact that the trimming cells can be updated on a time scale much shorter than the characteristic charging time of the UVPI devices; the trim charge is the integral of the stream of correction bits, and will settle to within a tiny neighborhood of the ideal quantity.

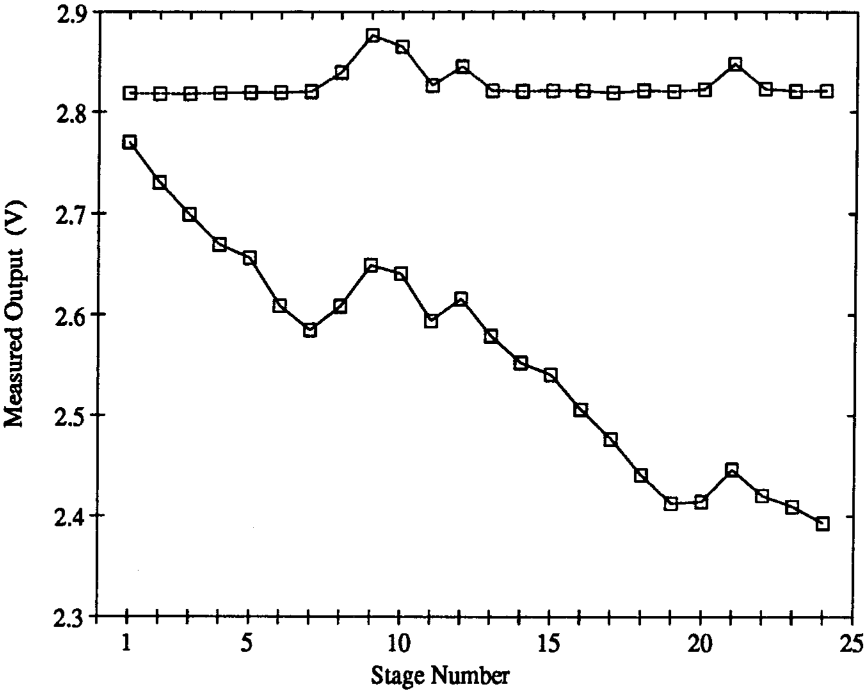


Figure 4.19: **Global offset trimming data:** *before and after*

Figure 4.19 plots the output voltages of each stage in the delay chain before and after the trimming operation. Before trimming, the chip was exposed to high–intensity shortwave UV for one hour, with a power supply of 0 V, in order to ensure an initial state of zero charge on the trimming nodes. The initial voltages on the delay chain show an average negative trend, as designed, but an occasional section has a mismatch large enough to overcome the designed negative offset and give a positive offset. These positive offsets cannot be corrected by the simple trimming circuitry, and so remain unchanged at the end of the trimming process.

Omission of the pullup transistor M3 in the circuit of Figure 4.18 would have enabled the trimming process to eliminate offsets of both signs, but would have rendered the circuit untestable in the event of a flaw in the trimming circuitry. The transistor M3 ensures a nonzero minimum pull–up current to the output node, and the trimming

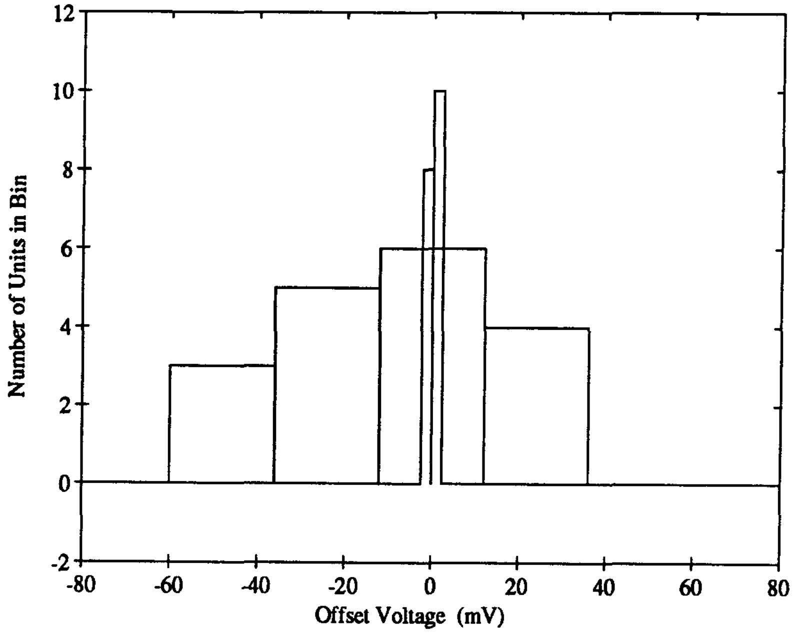


Figure 4.20: **Offset Reduction using the global trim method:** *The standard deviation of random offsets is reduced from 27mV before trimming to 790μV after trimming.*

device M4 can only increase the pull-up current from this minimum. A positive offset indicates that the pull-up current needs to be *reduced*, which is only possible if M3 is eliminated from the circuit.

If the un-correctable sections are ignored, we can see that the global trimming method makes a substantial improvement in the matching of the sections in the delay line. Figure 4.20 plots histograms of the random offsets before and after trimming. The standard deviation is reduced from 27 mV, which is typical of small MOS differential amplifiers in this technology, down to 790 μV, which is about the same as the noise floor of the amplifiers.

A similar technique can be applied to equalize current sources, allowing both offset and gain errors to be greatly reduced in a VLSA array. The current source equalization requires a current-mode comparator. Several different designs are possible for such a device. One of the most simple and reliable of these is a combination of a transimpedance amplifier (see Section 5.2) and a voltage–mode comparator. The system–level construction of a current source equalizer is identical to the voltage offset equalizer discussed above.

Chapter 5

Assorted Building Blocks

During the course of my time at Caltech, I found it either necessary or interesting to develop a variety of different building–block circuits. This chapter is meant as a catch–all for the assortment of odds and ends.

5.1 Conductances and transconductances

Conductance and transconductance circuits are generally useful building blocks for many different types of analog information processing systems. Conductances with various nonlinearities are useful for spreading–function networks such as the resistive network in Mahowald's silicon retina [4], the segmenting network of Koch and Luo [19], and the segmenting fuse network of Harris [18]. Transconductances are useful for the construction of OTA–C circuit blocks, for voltage–to–current conversion, and as components in conductance–transconductance networks.

5.1.1 sinh resistor

A minor modification of Mead's horizontal resistor circuit yields a conductance with an adjustable-scale $\sinh(\cdot)$ nonlinearity, rather than a $\tanh(\cdot)$ nonlinearity as in the original circuit ([1], chapter 7).

If the pass–transistors of Mead's resistor circuit are placed in parallel, rather than

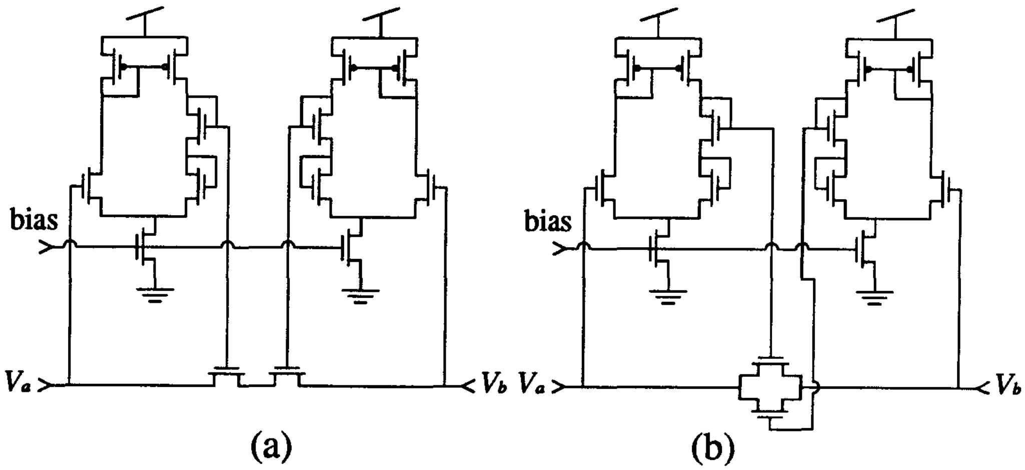


Figure 5.1: **Nonlinear resistor circuits:** *(a) Mead's horizontal resistor (HRes) [1], which gives a tanh nonlinearity, and (b) a simple modification of the HRes, which gives a sinh nonlinearity.*

series, as shown in Figure 5.1, a straightforward analysis shows that the current through the device is a sinh(·) function of the voltage across it. This circuit preserves the passivity characteristic of the original resistance circuit, while giving a transfer characteristic with a "hardening" nonlinearity rather than a "softening" nonlinearity. Such a difference can find use in a network which would *prevent* segmentation, or in time–domain circuits such as those discussed in Chapter 3. There is a substantial increase in linearity near the origin as the sinh–resistor circuit is operated above threshold. This linearity comes from the change in transistor characteristics from exponential to quadratic, and the accompanying qualitative change to a much milder nonlinearity.

Figure 5.2 plots measured characteristics of the circuit of Figure 5.1(b) for various bias settings. The intrinsic zero–offset nature of the circuit is clearly visible in the measured data. The signal path is purely passive, and all device mismatches are manifest as deviations from odd symmetry, rather than origin shifts.

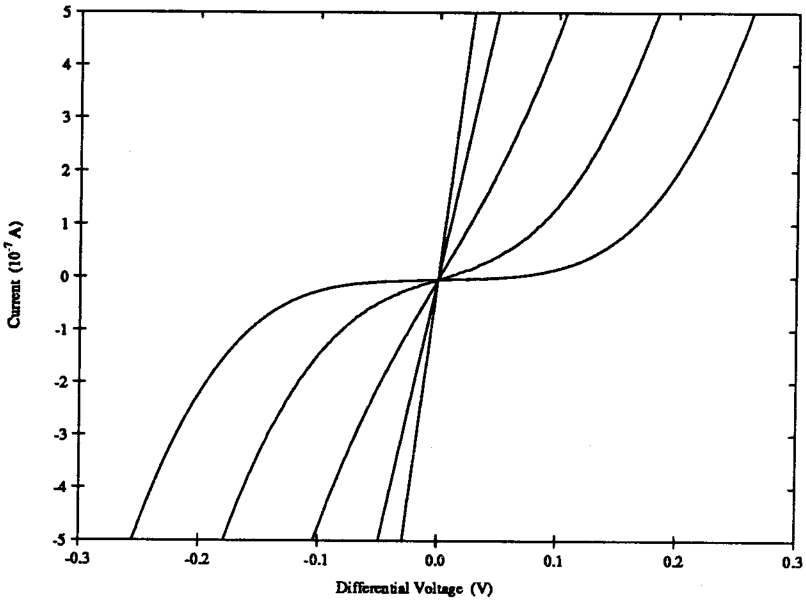


Figure 5.2: **Measured characteristics of the circuit of Figure 5.1(b)**

5.1.2 Arreguit's Early–effect amplifiers

Arreguit proposed a design for linear transconductance amplifiers which would give linear operation even for large input signals [43]. His original design was based on lateral bipolar transistors [44], but it is possible to design an all–MOS amplifier of the same type.

lateral bipolars

The simple linear transconductance amplifier originally proposed by Arreguit is diagrammed in Figure 5.3. It is possible to trim the amplifier by adjusting the voltages on the MOS gates M_1 and M_2 which separate the emitter diffusions from the collector diffusions. Using either the UV techniques discussed in Chapter 4 or hot–carrier or tunnelling techniques, one can design the circuit with floating MOS gates which can be trimmed to optimize circuit operation.

The layout of a UV–trimmed Arreguit amplifier is plotted in Figure 5.3, and test

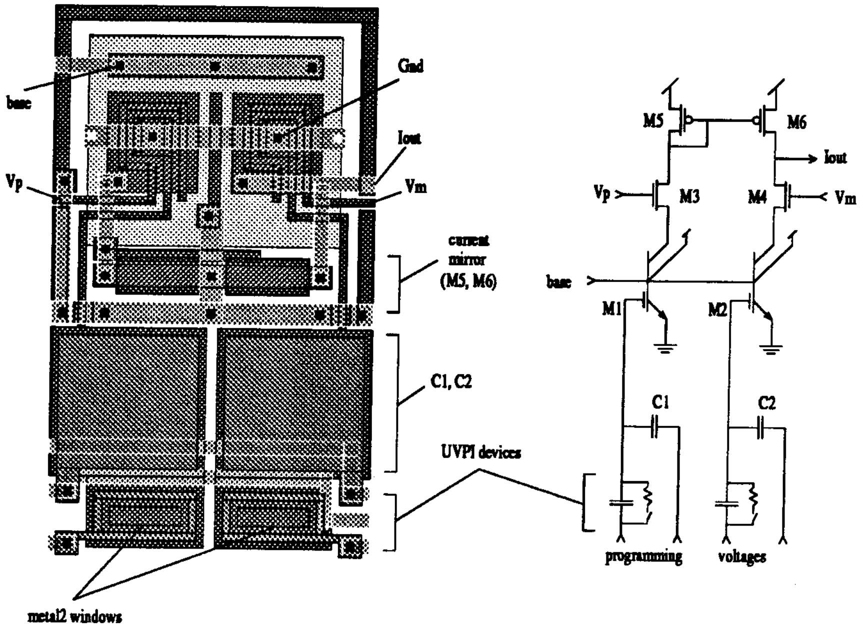


Figure 5.3: **Early–effect amplifier using lateral bipolars:** *Using CMOS compatible lateral bipolar transistors [44] as the conductance devices allows use of the MOS gate to trim out the offset voltage.*

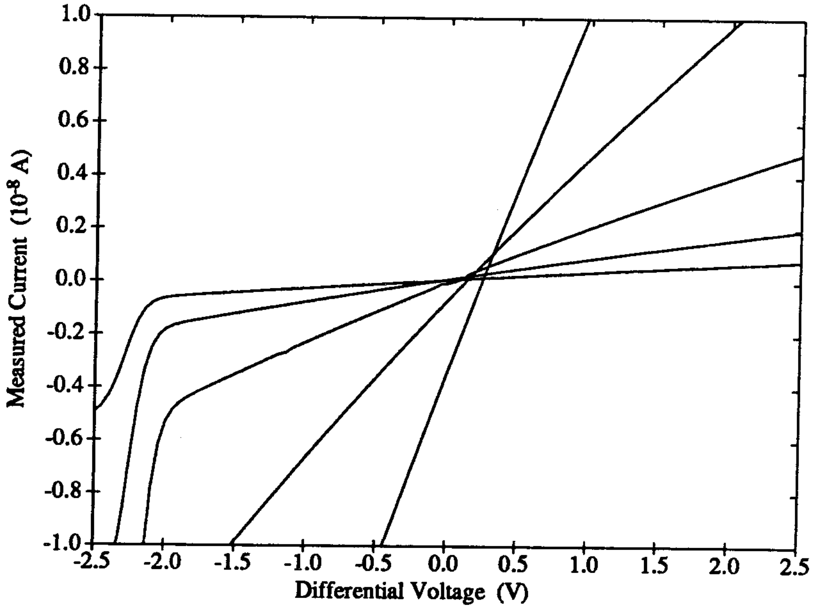


Figure 5.4: **Measured characteristics of the amplifier of Figure 5.3**

data is plotted in Figure 5.4.

MOS devices

The Early effect can be used just as easily in MOS devices as in bipolars, and MOS devices tend to occupy a much smaller area than bipolars. The circuit of Figure 5.5(a) implements an all–MOS version of Arreguit's amplifier. Amplifiers of this type show offsets of 100–500 mV. The all–MOS amplifier can be trimmed by the addition of a few transistors, as diagrammed in Figure 5.5(b); this circuit is a transconductance amplifier with two differential inputs, one high–gain ($V_{tp} - V_{tm}$), and one low–gain ($V_p - V_m$). In the case where the amplifier is intended for use with large signals, the high–gain input would be used to trim out the offset. This amplifier is also topologically very similar to a standard transconductance amplifier. If M4 were eliminated from the circuit of Figure 5.5(b), the transistor M3 could be used to trim out amplifier offsets using the Early effect on transistor M1, giving a trimmable transconductance amplifier requiring only a single additional transistor.

5.2 Transimpedances

Transimpedances are useful for current–to–voltage conversion operations. It often happens that it is especially convenient to represent a quantity as a current in one part of a system, and as a voltage in another part. For example, the global current trimming system of Section 4.5 could use a high–gain transimpedance as the comparator. Another example of a transimpedance circuit is in a CTCS implementation of a neural network. It would be sensible to represent the results of the synaptic computations as currents, so that the summation operation into the neuron is accomplished neatly by a single wire. It is then convenient to represent the neuron output as a voltage, so that it may

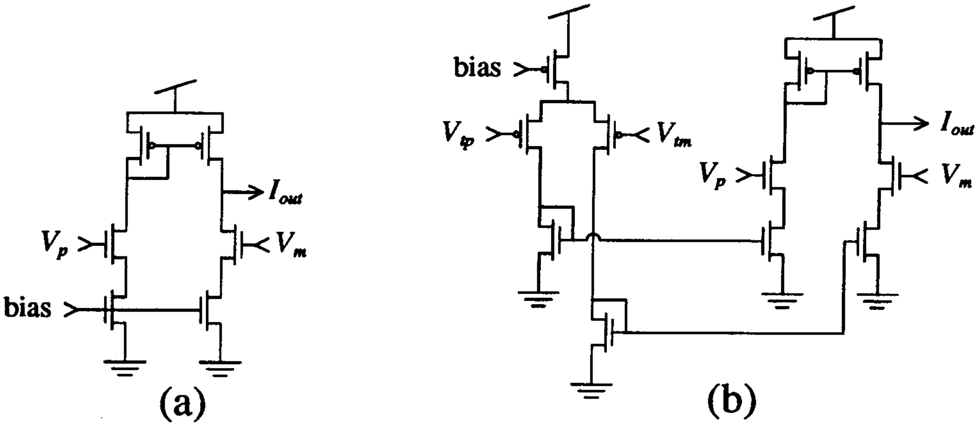


Figure 5.5: **An all–MOS Early–effect amplifier:** *(a) a simple translation of Arreguit's amplifier, and (b) a trimmable version. The mismatch between the Early-effect devices M1 and M2 can be trimmed out by the differential voltage ($V_{tp} - V_{tm}$) in this all-MOS circuit.*

be easily broadcast to a large array of synapses on a single wire. A "neuron" in such an implementation is thus a (possibly nonlinear) transimpedance element.

5.2.1 Inverse tanh

A classic method for constructing a transimpedance is to place a conductance or a transconductance in a feedback loop around an operational amplifier. When a simple tanh(·) transamp is used, the transimpedance resulting has an inverse tanh(·) nonlinearity, which goes rapidly to infinity for large arguments. Such a nonlinearity may be good in some circuits, but certainly $\tanh^{-1}(\cdot)$ is the wrong type of nonlinearity for neural networks.

5.2.2 Inverse sinh

If one replaces the feedback element with a sinh(·) circuit, the transimpedance has a $\sinh^{-1}(\cdot)$ nonlinearity, which, although unbounded, is generally sigmoidal, and useful for most neural network implementations of the sort mentioned above. Figure 5.7 plots

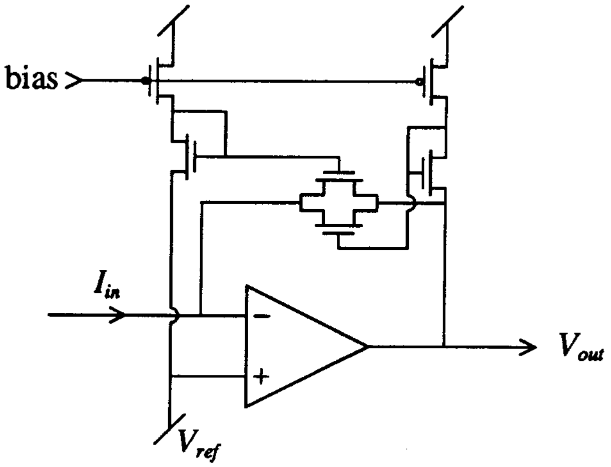


Figure 5.6: **Inverse sinh circuit:** *the sinh element is used in a feedback loop to yield an inverse sinh characteristic.*

experimental data from the circuit of Figure 5.6.

5.2.3 Linearized

If a linear transimpedance is required, the circuit designer has several options available. One can use the $\tanh^{-1}(\cdot)$ circuit with a capacitive linearization network, as discussed in Section 4.3. One can use a linear resistance, though resistances cost large areas in most fabrication processes. Using a current–divider network (a "T–network") in the feedback loop of an op–amp tends to reduce the area cost of using a linear resistance, at a great energy cost. Finally, one can use a linear transconductance, such as the amplifiers discussed in Section 5.1.2.

5.3 Bias generation

In many circuits, it is important to provide bias currents for various kinds of amplifiers. Often the circuits are insensitive to the exact magnitude of the bias current, and it is convenient for the bias to be generated on–chip by some subcircuit. It is often convenient to use a transistor to provide the required bias current, and think in terms of a bias

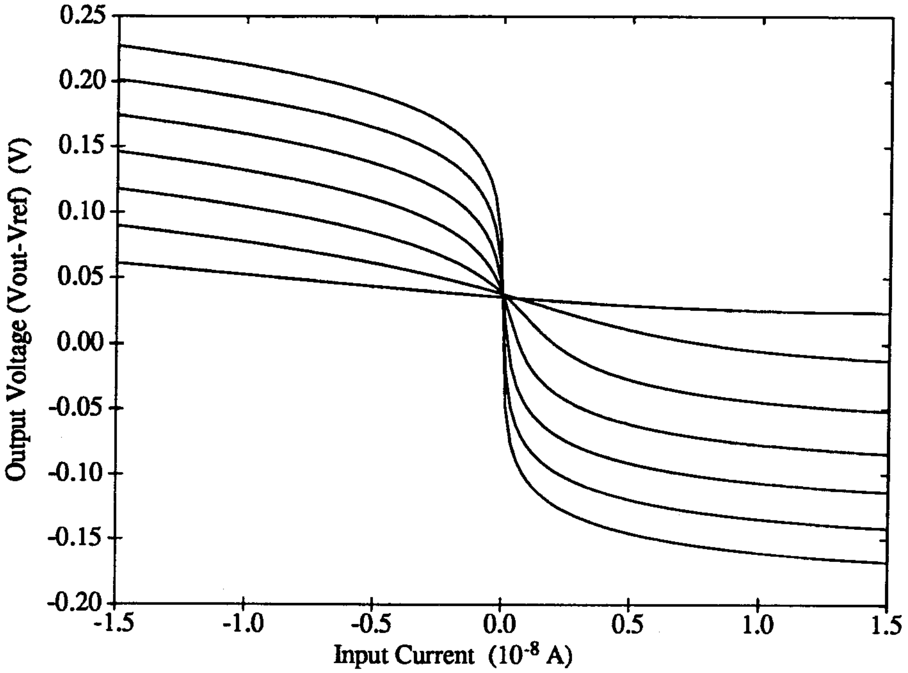


Figure 5.7: **Experimental data from the circuit of Figure 5.6:** *The inverse sinh characteristic can be scaled over several orders of magnitude.*

voltage supplied to the gate of the transistor. In this way, a bias can be broadcast to a group of circuits by means of a single wire.

5.3.1 Voltage dividers, V_{inv} bias

A common method for generating a low-precision reference voltage, either for biasing amplifiers or for a mid-supply reference level, is to use a stack of diode-connected MOS transistors across the power supply. Such diode stacks are voltage divider circuits, although they are somewhat nonlinear. In the case of the V_{inv} reference, illustrated in Figure Vi(a), the nonlinearity is such that the output voltage is rather insensitive to the load current.

A disadvantage to using diode stack reference sources is that they tend to consume a rather large static current. In many applications, the reference voltage will simply be attached to a transistor gate, so no static current drain is really necessary. In such

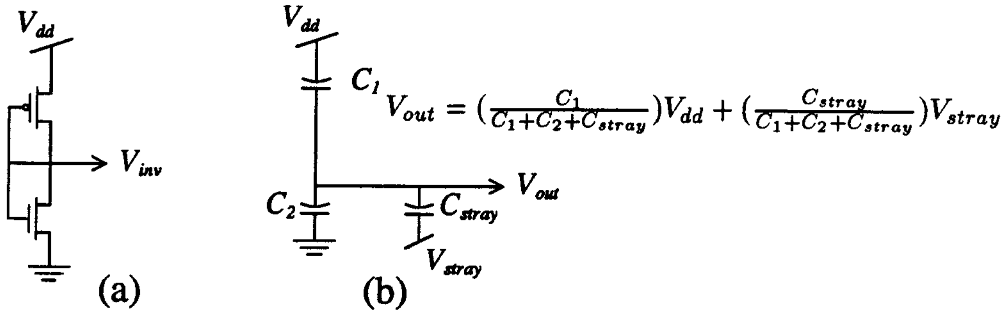

V_{dd}
V_{inv}
(a)
V_{dd}
C_1
$V_{out} = (\frac{C_1}{C_1+C_2+C_{stray}})V_{dd} + (\frac{C_{stray}}{C_1+C_2+C_{stray}})V_{stray}$
V_{out}
C_2
C_{stray}
V_{stray}
(b)

Figure 5.8: **Voltage divider references:** *(a) a simple diode stack to give V_{inv}, the inverter threshold, and (b) a capacitive voltage divider from the power supply. Note that stray capacitance must be carefully designed into the circuit.*

cases, capacitive voltage dividers may be used to provide reference levels with zero power cost. A capacitive reference circuit is diagrammed in Figure 5.8(b). Notice that stray capacitances must be designed into such a circuit; in order for the capacitive divider to operate as designed, there can be no substantial "stray" capacitances. In addition, the total charge on the floating node V_{ref} must be set in some way. A very simple scheme is simply to expose the chip to UV radiation without any power applied, so that the node charge is zero. Nonzero node charges can occur in such a scheme, even after UV exposure, if the doping types of the floating node and of nearby silicon areas are substantially different. For details on the (p-Si)–SiO$_2$–(n-Si) structure, see the experimental results of Section 4.1.

5.3.2 V_e bias

It is possible to generate a reference voltage using the different characteristics of MOS and bipolar transistors. Below threshold, a MOS transistor has a saturation current that varies exponentially in gate voltage. Bipolar transistors, similarly, follow exponential characteristics, but the exponent is different in the MOS transistor case than in the

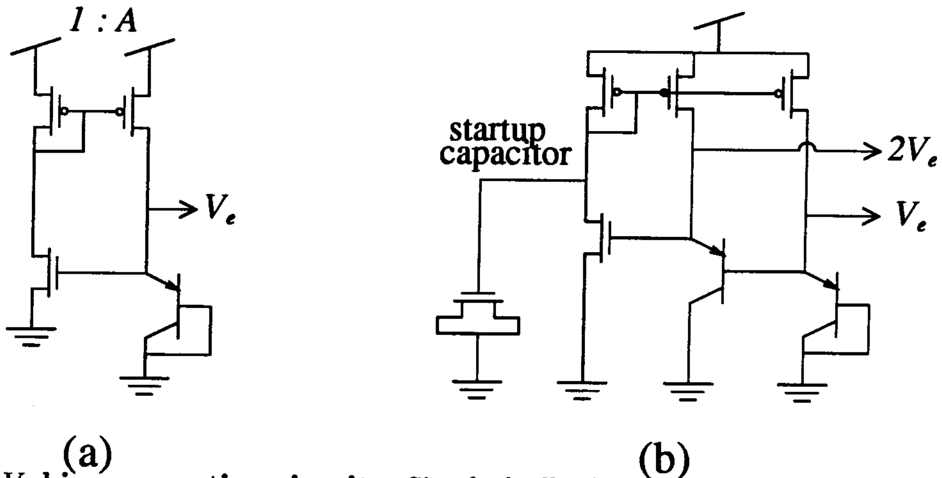


Figure 5.9: V_e **bias generation circuit:** *Simple feedback loops around MOS and bipolar transistors allow the generation of a reference voltage that depends on the difference between MOS and bipolar characteristics.*

bipolar case. It is possible to construct a feedback loop around a pair of devices to extract the voltage at which the two types of transistors have identical currents. This voltage is often referred to as V_e in analog design texts, as the base–emitter voltage of a bipolar transistor varies only logarithmically with current, and so can be considered nearly constant over a wide range.

It is possible to use the parasitic vertical bipolar transistor in a bulk CMOS process to build such a circuit, without requiring a special BiCMOS fabrication process. Such a circuit is diagrammed in Figure 5.9(a). The current gain aroud the loop must exceed unity at low currents, otherwise the circuit will only settle to a zero–current condition. This loop gain requirement can be met by the width of the n-transistor and/or by choosing an appropriately large mirror ratio A. In addition, there should be some provision for avoiding the zero–current condition at startup time, even if the loop gain is sufficient to render the zero–current operating point unstable. Operation near zero current will be extremely slow, so departure from the zero–current condition should be ensured by some kind of startup circuit, such as the capacitor in the circuit of Figure 5.9(b).

In order to achieve a large loop gain, the simple circuit of Figure 5.9(a) becomes prohibitively large, with $A \approx 100$. A much larger loop gain can be achieved by using a stack of base–emitter voltages, as in the circuit of Figure 5.9(b). This circuit has the disadvantage that, for reasonable device ratios, the zero–current operating point is *stable*, so the startup capacitor is absolutely necessary.

Chapter 6

Future Directions

Where might one go from here? An obvious direction is to refine the use of the techniques presented in essentially the same ways presented. Such incremental improvements as better choice of UV light sources, better control of light leakage, new delay–line unit circuits will all doubtless prove to be useful.

Structures similar to those used for UV detection may lend themselves to detection and dosimetry of higher–energy ionizing radiation. Integration of a detector with calibration and readout electronics could allow the construction of low–cost, general–purpose "smart" dosimeters.

Far more exciting, however, is the possibility of using this work in analog circuit design to jump off in completely new directions. Two particularly interesting possibilities present themselves to me: (1) using hybrid information representations to take more advantage of the low–level physics of circuit elements, while still preserving the advantages of digital representations, and (2) designing "intelligent" IC fabrication processes, in which the circuit under fabrication partially controls the fabrication process.

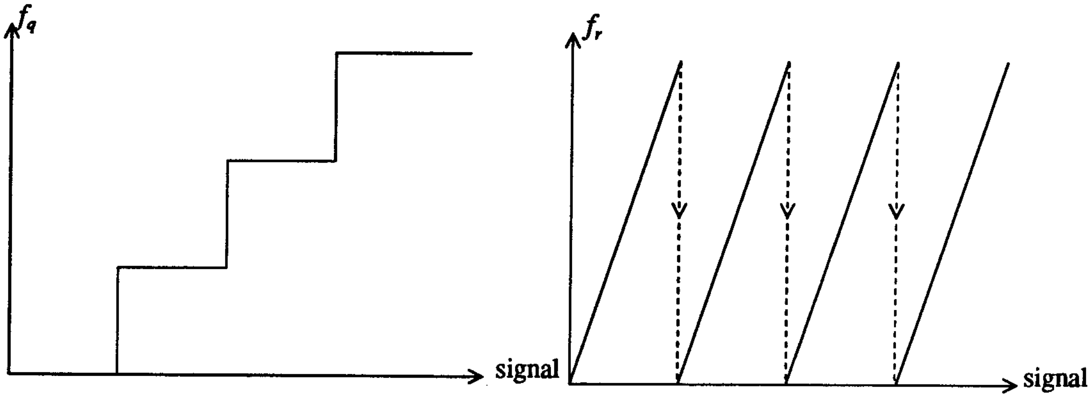


Figure 6.1: **transfer functions of a subranging ADC section:** *a single input quantity is distributed over two wires according to the two functions f_q and f_r. The residue function f_r is discontinuous.*

6.1 Hybrid representations

As mentioned in Chapter 2, logarithmic digital representations are very efficient for high–precision and/or high–speed computations. Some analog computations make use of logarithmic transformations or other nonlinear transformations in order to facilitate certain computational operations, but these representations are typically still single–wire signals. It should be possible to define a continuous or piecewise continuous representation that distributes a single signal over many wires, thus sidestepping the fundamental signal–to–noise limitation found on a single–wire representation.

One example of such a distributed representation is found in subranging analog–to–digital converters. In each stage of a subranging ADC, the incoming analog signal is converted to a discrete–valued portion and a continuous–valued residue portion. The continuous–valued residue can then be piped to another stage. The typical binary encoding scheme used in most ADCs results in a discontinuous residue function, as illustrated in Figure 6.1. The discontinuities in the residue function imply that small

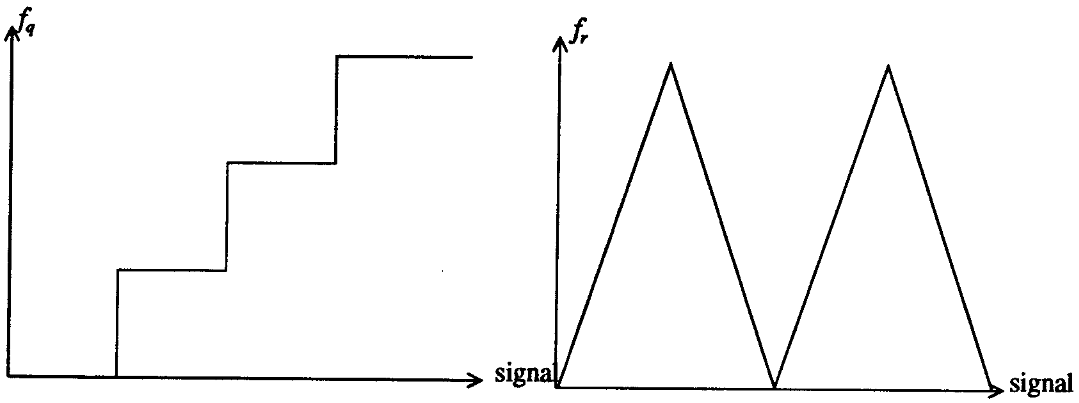


Figure 6.2: **Continuous ADC transfer functions:** *A simple alteration of f_r gives a continuous function.*

changes in the input can result in extremely large changes in the output: enormous gain. Hence, a discontinuous representation can cause serious circuit design problems. A conceptually simple fix for this gain problem is to ensure that the residue function is continuous. Figure 6.2 diagrams a simple alteration of the strict binary encoding scheme in such a way that discontinuities are avoided.

In general, one would like to avoid discontinuities in any multiwire encoding scheme, so that small changes in the input always result in relatively small changes in the outputs. Any given multiwire encoding can be considered to be a curve in N–space, where N is the number of wires on which the signal is to be represented. In the case of the simple ADC stage discussed above, one axis of our space is discrete–valued, while the other is continuous–valued. The two encoding schemes of Figures 6.1 and 6.2 are drawn in 2–space in Figure 6.3. With this picture, it is easy to see that one can construct an infinite variety of multiwire encoding schemes simply by specifying various curves in spaces of the desired dimensions.

Multiwire representations of signals might be useful in performing continuous–valued

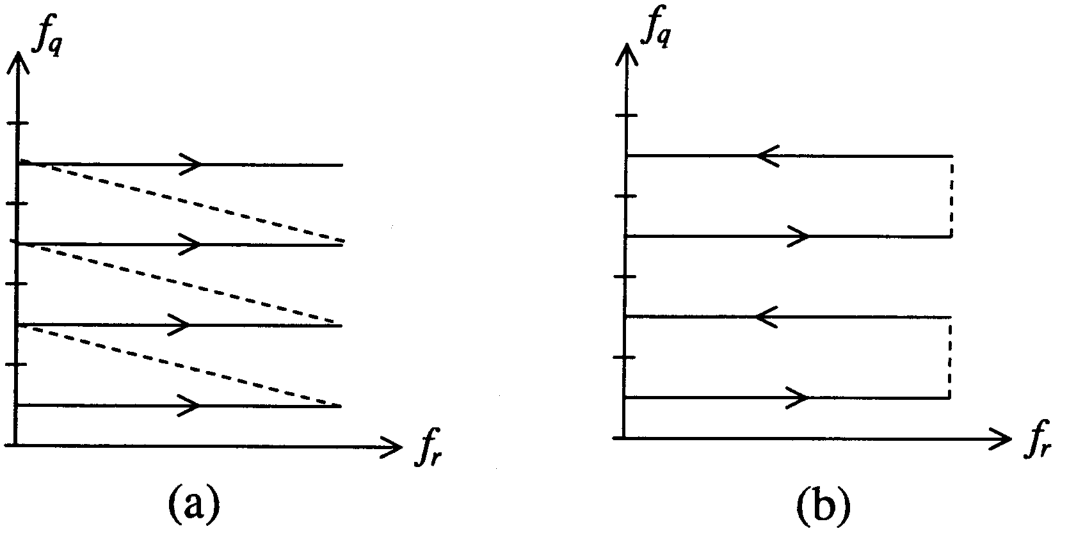


Figure 6.3: **Space–curve representation of ADC stages:** *Any single–wire to multiwire encoding scheme can be represented by a one–dimensional curve in the output space. Examples are (a) the encoding scheme of Figure 6.1, and (b) that of Figure 6.2.*

computations at higher precision than is possible on a single wire due to noise limitations. A multiwire analog signal representation could share the logarithmic scaling properties of standard digital codes and the space and energy efficiency of standard analog representations. A simple single–wire to multiwire encoder can be constructed from a sine–function circuit [59] or similar smooth non–monotonic function circuit, in a way very similar to a subranging ADC section.

6.2 Fabrication processes

The automatic offset correction technique described in Chapter 4 can be considered a final fabrication step for the circuits. This final step uses locally constructed feedback loops to correct for component variations in the initial fabrication of the circuits. The correction process might use some circuitry that exists solely for the trimming process, and is never again used in the normal operation of the chip. Inclusion of throw–away circuitry is perfectly acceptable if it manages to increase the overall system performance by reducing manufacturing costs or increasing component reliability.

It seems that it may be profitable to try to push an active correction step further back into the fabrication process, perhaps by constructing active circuitry which will be included in the control loop of the fabrication machinery. In this way, earlier layers of circuitry might test and guide the construction of later layers, allowing greatly increased yields by actively correcting defects. A closed-loop fabrication process would be especially practical and useful for multilayer circuits of the type reported by Kioi *et al.* [58]. The idea of closed-loop fabrication processes is not entirely new. Burggraaf proposed to continuously map wafer properties using scanning electron microscopy and feed the data back into the fabrication process to avoid defects [60].

It is interesting to note that biological systems also use a certain amount of throw-away circuitry during brain development. Neurobiologists are as yet unsure of the function of much of the throw-away structures, but it appears that they are essential to correct brain development. Perhaps the semiconductor industry could profit from a trade of ideas with the biologists.

References

[1] C. A. Mead,
Analog VLSI and Neural Systems,
Addison–Wesley Publishing, Reading, MA USA, 1989.

[2] C. A. Mead, L. Conway,
Introduction to VLSI Sytems,
Addison–Wesley Publishing, Reading, MA USA, 1980.

[3] C. A. Mead,
Adaptive Retina,
in Carver Mead and Mohammed Ismail, ed., *Analog VLSI Implementation of Neural Systems,* pp.239–246. Kluwer Academic Publishers, Norwell, MA USA, 1989.

[4] M. Mahowald, C. Mead,
The silicon retina,
Scientific American, vol.264, no.5, pp.76–82 1991.

[5] T. Horiuchi, private communication. There is activity both at Caltech and at the Rockwell Research Center to use silicon retinas and similar imager/processors as front ends for visually driven robotic vehicles.

[6] Lothar Lerach,
LSI/VLSI for Telephony,
chapter 13 in *Design of MOS VLSI Circuits for Telecommunications,* Y. Tsividis and P. Antognetti, ed., Prentice–Hall, Inc., Englewood Cliffs, NJ USA, 1985.

[7] William G. Wolber, Kensall D. Wise,
Sensor Development in the Microcomputer Age,
IEEE Transactions on Electron Devices, vol.26, no.12, pp.1864–1874, December, 1979.

[8] R. F. Lyon, C. A. Mead,
An Analog Electronic Cochlea,
IEEE Transactions on Acoustics, Speech, and Signal Processing, vol.36, no.7, pp.1119–1134, July 1988.

[9] R. F. Lyon,
Analog Implementations of Auditory Models,
DARPA Workshop on Speech and Natural Language, Morgan Kaufmann Publishers, San Mateo CA, 1991.

[10] J. P. Lazzaro,
Silicon Models of Early Audition,
California Institute of Technology PhD thesis, 1990.

[11] D. L. Watts,
California Institute of Technology PhD thesis, 1993.

[12] S. Ryckebusch, J. M. Bower, C. Mead,
Modeling small oscillating biological networks in analog VLSI,
Advances in Neural Information Processing Systems I, pp.384–393. David S. Touretzky, ed., Morgan Kaufmann Publishers, San Mateo CA USA 1989.

[13] Misha Mahowald, Rodney Douglas,
A silicon Neuron,
Nature, vol.354, no.19, pp.515–518, 26 December 1991.

[14] Donald M. Mackay, Michael E. Fisher,
Analogue computing at ultra-high speed; an experimental and theoretical study, John Wiley, New York, 1962.

[15] R. Tomovic, W. J. Karplus,
High-speed analog computers, John Wiley publishing, New York, USA, 1962.

[16] M. A. Sivilotti,
Wiring considerations in analog VLSI systems, with application to field–programmable networks,
California Institute of Technology PhD thesis, 1991.

[17] V. Braitenberg,
Vehicles — Experiments in Synthetic Psychology, MIT Press, 1984.

[18] John G. Harris,
Analog Models for Early Vision,
California Institute of Technology PhD thesis, 1992.

[19] J. Hutchinson, C. Koch, J. Luo, C. Mead,
Computing motion using analog and binary resistive networks,
IEEE Computer,, March, 1988, pp.52–63.

[20] , J. Luo, C. Koch, C. Mead,
Analog VLSI circuits for surface interpolation,
Poster presentation at *Neural Information Processing Systems,* Nov. 28 – Dec. 1, 1988, Denver, CO USA.

[21] A. Pavasović,
Subthreshold Region MOS Mismatch Analysis and Modeling for Analog VLSI Systems,
Johns Hopkins University PhD thesis, 1990.

[22] J. J. Hopfield,
Effectiveness of Analogue "neural network" hardware,
Network: computation in neural systems vol.1, no.1, pp.27–40, January, 1990.

[23] E. Vittoz,
Future trends of analog in the VLSI environment,
Proceedings of the IEEE International Symposium on Circuits and Systems 1990 vol.2, pp.1372–1375. IEEE International Symposium on Circuits and Systems, New Orleans, LA USA, May 1–3 1990. IEEE Press, Piscataway, NJ USA, 1990.

[24] B. J. Hosticka, W. Brockherde,
The art of analog circuit design in a digital VLSI world,
Proceedings of the IEEE International Symposium on Circuits and Systems 1990 vol.2, pp.1347–1350. IEEE International Symposium on Circuits and Systems, New Orleans, LA USA, May 1–3 1990. IEEE Press, Piscataway, NJ USA, 1990.

[25] B. J. Hosticka,
Performance comparison of analog and digital circuits,
Proceedings of the IEEE, vol.73, no.1, pp.25–29, January, 1985.

[26] M. Banu and Y. Tsividis,
Floating Voltage–Controlled Resistors in CMOS Technology,
Electronics Letters, vol.18, no.15, pp.678–679 (1982).

[27] Mohamed E. El-Hawary,
Control System Engineering, Reston Publishing, Reston, Virginia, 1984. Section 6.3 concerns the Routh-Hurwitz stability criterion.

[28] Paul R. Gray and Robert G. Meyer,
Analysis and Design of Analog Integrated Circuits, second edition. John Wiley & Sons, New York, 1984.

[29] N. W. McLachlan,
Ordinary Nonlinear Differential Equations in Engineering and Physical Sciences, Clarendon Press, Oxford, 1950.

[30] Ali H. Nayfeh, Dean T. Mook,
Nonlinear Oscillations, John Wiley & Sons, 1979.

[31] Willy Sansen,
Analog Functional Blocks,
Swiss Federal Institute of Technology Intensive Summer Course on CMOS VLSI Design, 1989. Especially relevant is the section on the Symmetrical CMOS OTA.

[32] R. Senani,
New electronically tunable OTA-C sinusoidal oscillator,
Electronics Letters, vol.25, no.4, pp.286–287 (1989).

[33] M. Steyaert, P. Kinget, W. Sansen,
Full Integration of Extremely Large Time Constants in CMOS,
Electronics Letters, vol.27, no.10, pp.790–791 (1991).

[34] L. Richard Carley,
Trimming Analog Circuits Using Floating-Gate Analog MOS Memory,
IEEE Journal of Solid-State Circuits, vol.24, no.6, pp.1569–1575, December 1989.

[35] Lance A. Glasser,
A UV Write–Enabled PROM,
in Henry Fuchs, ed., *Chapel Hill Conference on VLSI (1985),* Computer Science Press, Rockville, MD, 1985, pp.61–65.

[36] J. R. Mann,
Floating gate Circuits in MOSIS,
MIT Lincoln Labs Technical Report 824, 1 November 1990.

[37] R. J. Powell,
Interface barrier energy determination from voltage dependence of photoinjected currents,
Journal of Applied Physics, vol.41, no.6, pp.2424–2432, (1970).

[38] D. A. Kerns, J. E. Tanner, M. A. Sivilotti, J. Luo,
CMOS UV–Writable Non–Volatile Analog Storage,
in C. Séquin, ed., *Advanced Research in VLSI: proceedings of the UC Santa Cruz Conference 1991,* MIT Press, Cambridge, MA, 1991, pp.245–261.

[39] Lewis R. Koller,
Ultraviolet Radiation, second edition, John Wiley and Sons, New York, 1965.

[40] *Biological Impacts of Increased Intensities of Solar Ultraviolet Radiation,* A report of the *ad hoc* panel on the biological impacts of increased intensities of solar ultraviolet radiation to the Environmental Studies Board of the National Academy of Sciences (USA), 1973, Washington, DC.

[41] Gad Shani,
Radiation Dosimetry Instrumentation and Methods, CRC Press, Boca Raton, FL USA, 1991. Chapter 8 is devoted to solid–state detectors and dosimeters.

[42] R. G. Benson, D. A. Kerns,
UV Activated Conductances Allow Multiple Time–Scale Learning,
IEEE Transactions on Neural Networks, in press, 1992.

[43] X. Arreguit, private communication: Dr. Arreguit proposed turning a vice of the CLBT (small Early voltage) into a virtue by using the proper circuit topology to use the small transconductance.

[44] Xavier Areguit,
Compatible Lateral Bipolar Transistors in CMOS Technology: Model and Applications,
Ecole Polytechnique Federale de Lausanne DSc thesis, These no.817, 1989.

[45] D. A. Kerns,
A Monolithic UV Detector–Dosimeter,
Sensors and Actuators A, in press.

[46] Carver A. Mead, Timothy P. Allen,
Subthreshold CMOS Amplifier with Offset Adaptation,
US Patent number 4,935,702. 1990.

[47] R. Tawel, R. Benson, A. Thakoor,
A CMOS UV–programmable non–volatile synaptic array,
Proceedings of the International Joint Conference on Neural Networks, Seattle, Washington, July 8–12, 1991. vol.1, pp.581–585. IEEE Press, Piscataway, NJ USA, 1991.

[48] L. Watts, D. Kerns, C. Mead, R. Lyon,
Improved Implementation of the Silicon Cochlea,
IEEE Journal of Solid-State Circuits vol,27, no.5, pp.692–700, May 1992.

[49] Eduard Säckinger, Walter Guggenbühl, An Analog Trimming Circuit Based on a Floating–Gate Device, *IEEE Journal of Solid-State Circuits,* vol.23, no.6, pp.1437–1440, December, 1988.

[50] C. Mead, T. Delbrück,
Scanners for visualizing activity of analog VLSI circuitry,
Computational and Neural Systems Memo no.11, California Institute of Technology Computational and Neural Systems Library, 1991.

[51] Mary Ann Maher, private communication. The idea of using matched capacitors driven by complementary signals is simple, but not necessarily obvious, so credit should be given to the source.

[52] Daniel H. Sheingold, ed.,
Nonlinear Circuits Handbook, second edition Analog Devices, Inc., Norwood, Massachusetts USA 1976.

[53] E. Vittoz, H. Oguey, M. A. Maher, O. Nys, E. Dijkstra, and M. Chevroulet,
Analog storage of adjustable synaptic weights.
In U. Ramacher and U. Rückert, editors, *VLSI Design of Neural Networks*, pp.47–63. Kluwer Academic Publishers, 1991.

[54] D. J. DiMaria, F. J. Feigl, and S. R. Butler,
Capture and emission of electrons at 2.4 eV–deep trap level in SiO_2 films.
Physical Review B, vol.11, no.12, pp.5023–5030, 15 June 1975.

[55] Y. Nissan-Cohen, J. Shappir, and D. Frohman-Bentchkowsky,
Dynamic model of trapping-detrapping in SiO_2
Journal of Applied Physics, vol.58, no.6, pp.2252–2261, 15 September 1985.

[56] C. Carroll,
Hybrid Computation,
California Institute of Technology PhD thesis, 1981.

[57] G. B. Whitham,
Linear and Nonlinear Waves, John Wiley & Sons, New York, USA 1974.

[58] K. Kioi, T. Shinozaki, S. Toyoyama, K. Shirakawa, K. Ohtake, S. Tsuchimoto,
Design and implementation of a 3D–LSI image sensing processor,
IEEE Journal of Solid-State Circuits, vol.27, no.8, pp.1130–1140, August 1992.

[59] O. Ishizuka, Z. Tang, H. Matsumoto
MOS sine function generator using exponential–law technique,
Electronics Letters, vol.27, no.21, pp.1937–1939, 1991.

[60] P. Burggraaf,
What's in your implanter evaluation toolbox?,
Semiconductor International, vol.11, no.12, pp.77–83, November 1988.

[61] T. Rahkonen, J. Kostamovaara,
Pulsewidth measurements using an integrated pulse shrinking delay line,
Proceedings of the IEEE International Symposium on Circuits and Systems 1990 vol.1, pp.578–581. IEEE International Symposium on Circuits and Systems, New Orleans, LA USA, May 1–3 1990. IEEE Press, Piscataway, NJ USA, 1990.

[62] G. Teuchertnoodt, K. H. Breuker, R. R. Dawirs,
Neuronal lysosome accumulation in degrading synapses of sensory–motor and limbic subsystems in the duck anas-platyrhynchos — indication of rearrangements during avian brain–development,
Developmental Neuroscience, vol.13, no.3, pp.151–163, 1991.

Appendix A

Symbols and Conventions

Different texts and papers in the literature use varying symbols and pictorial conventions. This appendix lists the symbols and conventions used in the preceding text.

A.1 Symbols

k	Boltzmann's constant	1.380658×10^{-23} J-K^{-1}
T	Absolute temperature (Kelvins)	about 300 at room temp.
q	electron charge magnitude	$1.60217733 \times 10^{-19}$ C
U_T	thermal voltage (kT/q)	25.8 mV at 300 K
Λ_T	inverse thermal voltage	39.3 V^{-1} at 300 K

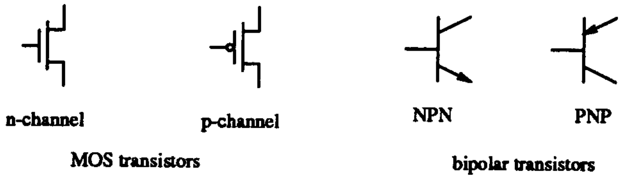
A.2 Naming conventions

Electrical quantities are named according to common convention: C for capacitances, g for conductances and transconductances, V for voltages, I for currents, q for charges. In addition, the following abbreviations are used throughout the text:

MOS	Metal Oxide Semiconductor (often, nowadays, Silicon–oxide–Silicon)
OTA	Operational Transconductance Amplifier
CTCS	Continuous–Time, Continuous–Signal
CTDS	Continuous–Time, Discrete–Signal
DTCS	Discrete–Time, Continuous–Signal
DTDS	Discrete–Time, Discrete–Signal
CLBT	CMOS Compatible Lateral Bipolar Transistor [44]

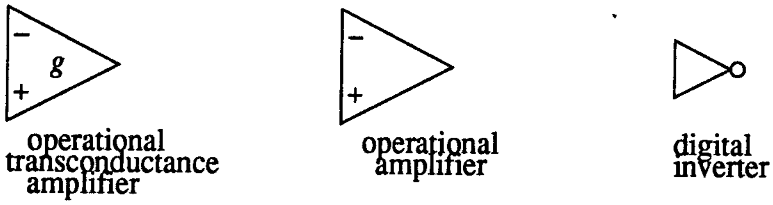
A.3 Pictorial conventions

A.3.1 Transistors



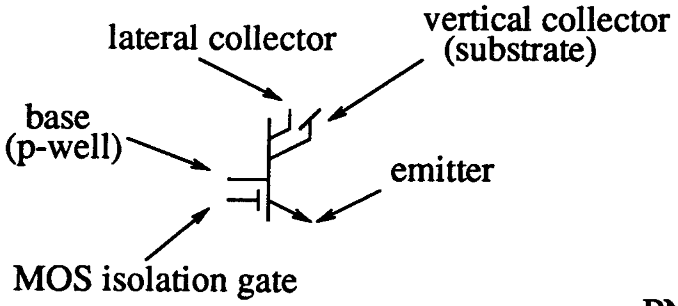
Simple symbols are used for MOS transistors, with a bubble on the p-channel devices. Bulk terminals of the transistors are assumed to be tied to the appropriate extremes of the power supply unless otherwise noted.

A.3.2 Amplifiers

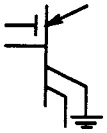


Standard differential amplifier symbols are used for operational amplifiers. The op–amp symbol with a g inside denotes an operational transconductance amplifier (the output *current* is the quantity of interest in the case of an OTA). A triangular symbol with a bubble on the output denotes a digital inverter constructed of a complementary pair of MOS transistors.

A.3.3 Miscellany



NPN CMOS-compatible lateral bipolar transistor



PNP CMOS-compatible lateral bipolar transistor (n-well process)

CMOS compatible lateral bipolars are schematized using Arreguit's symbol, which includes the MOS transistor gate used to define the base width, as well as an extra collector to denote the vertical collector action of the substrate in a CLBT fabricated in a CMOS bulk process.